US009477874B2

(12) **United States Patent**
Luo

(10) **Patent No.: US 9,477,874 B2**
(45) **Date of Patent: Oct. 25, 2016**

(54) **METHOD USING A TOUCHPAD FOR CONTROLLING A COMPUTERIZED SYSTEM WITH EPIDERMAL PRINT INFORMATION**

(71) Applicant: **Handscape Inc.**, Mountain View, CA (US)

(72) Inventor: **Tong Luo**, Fremont, CA (US)

(73) Assignee: **Handscape Inc.**, Fremont, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **14/746,628**

(22) Filed: **Jun. 22, 2015**

(65) **Prior Publication Data**

US 2016/0034738 A1 Feb. 4, 2016

Related U.S. Application Data

(63) Continuation-in-part of application No. 14/268,926, filed on May 2, 2014, and a continuation-in-part of application No. 14/260,195, filed on Apr. 23, 2014, now Pat. No. 9,430,147, and a continuation-in-part of application No. 13/770,791, filed on Feb. 19, 2013, now Pat. No. 9,311,724, and a continuation-in-part of application No. 13/223,836, filed on Sep. 1, 2011, now Pat. No. 9,310,905, said application No. 13/770,791 is a continuation-in-part of application No. 12/773,075, filed on May 4, 2010, now Pat. No. 8,384,683.

(60) Provisional application No. 62/016,026, filed on Jun. 23, 2014, provisional application No. 62/016,028, filed on Jun. 23, 2014, provisional application No. 61/819,615, filed on May 5, 2013, provisional application No. 61/815,058, filed on Apr. 23, 2013, provisional application No. 61/327,102, filed on Apr. 23, 2010.

(51) **Int. Cl.**

G06K 9/00	(2006.01)
G06F 3/0354	(2013.01)
G06F 3/0488	(2013.01)
G06F 3/041	(2006.01)
G06F 3/042	(2006.01)

(52) **U.S. Cl.**

CPC ***G06K 9/001*** (2013.01); ***G06F 3/03547*** (2013.01); ***G06F 3/0418*** (2013.01); ***G06F 3/0425*** (2013.01); ***G06F 3/0488*** (2013.01); ***G06K 9/0008*** (2013.01); *G06F 2203/0338* (2013.01); *G06F 2203/04104* (2013.01); *G06F 2203/04808* (2013.01)

(58) **Field of Classification Search**

None

See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

7,102,617 B2 *	9/2006	Gust	.....	G06F 3/0317 345/158
2010/0013777 A1 *	1/2010	Baudisch	.....	G06F 1/1626 345/173

* cited by examiner

Primary Examiner — Hadi Akhavannik

(74) *Attorney, Agent, or Firm* — Stephen E. Zweig

(57) **ABSTRACT**

Methods of processing touchpad data are disclosed. The touchpad data indicates one or more areas of the touch pad contacted by a user, and indicates print information indicating a pattern of epidermal ridges of the user in the contacted areas. The methods may also include analyzing the touchpad data in accordance with a model of a human hand. The methods may also include associating the content areas with particular anatomical portions, such as fingers, where the associating is based at least in part on the model, the contact areas, and the print information. The methods may also include modifying one or more parameters of the model based on the print information. The methods may also include obtaining print information from contact areas selected based on features associated with the contact areas.

17 Claims, 36 Drawing Sheets

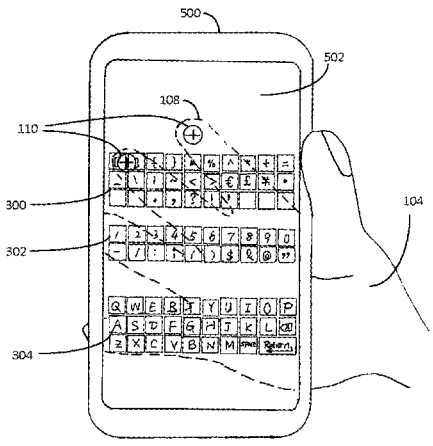


Figure 1

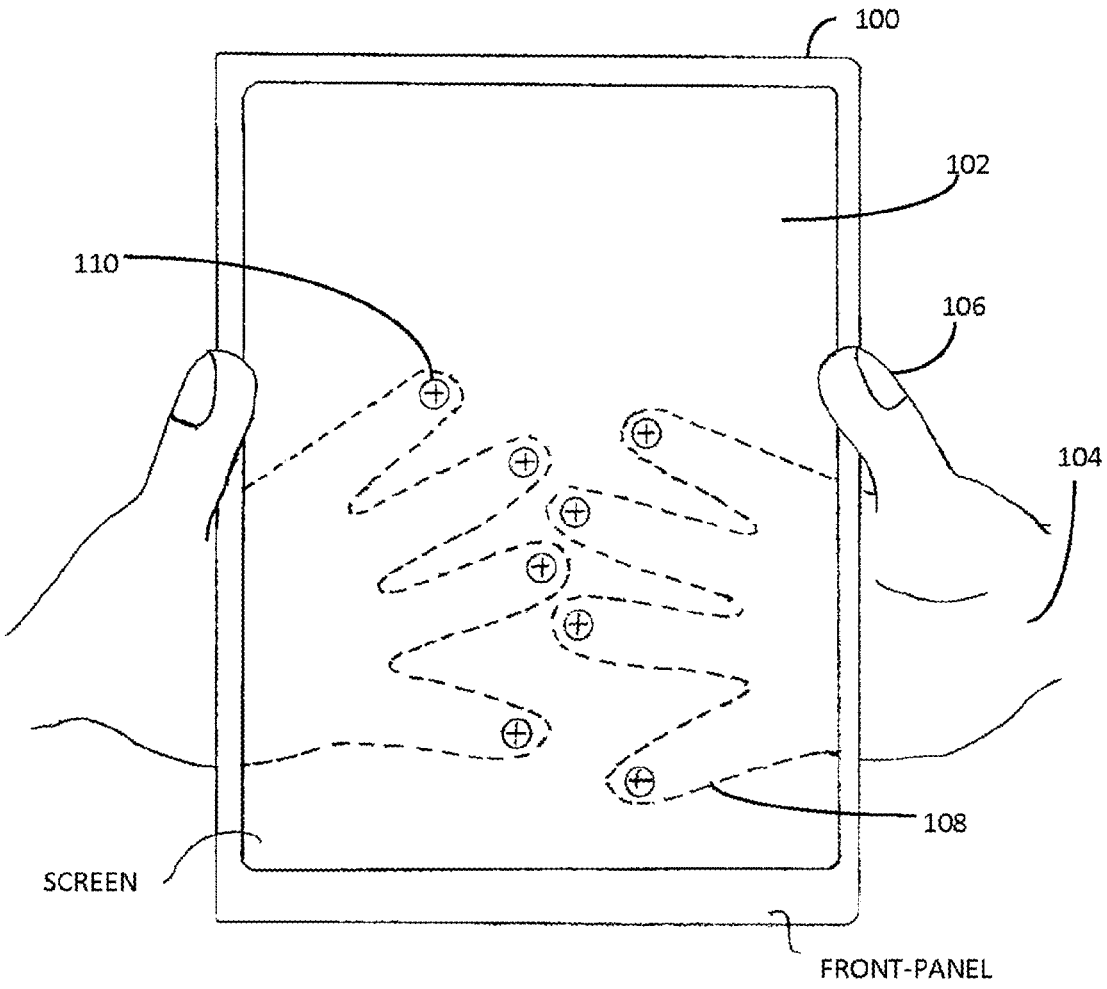


Figure 2

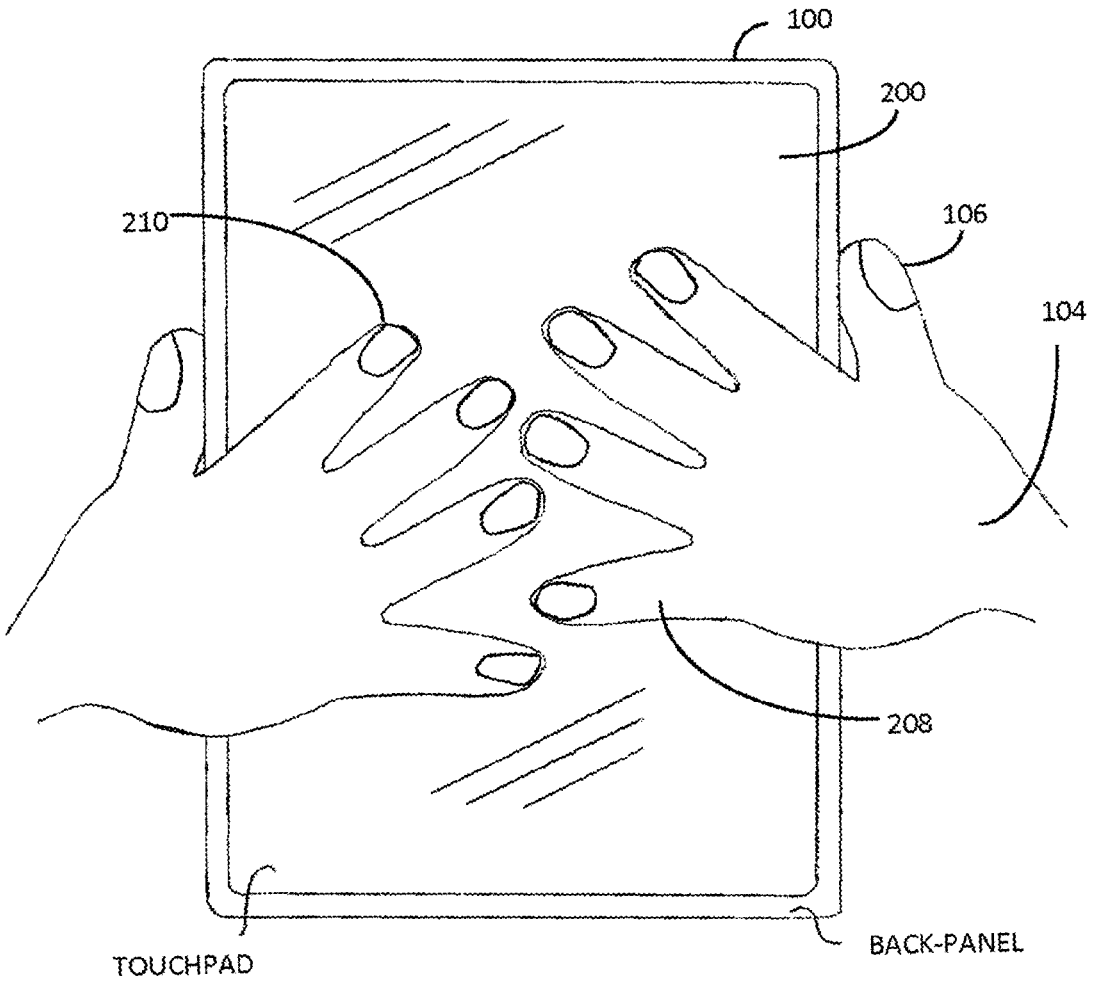


Figure 3

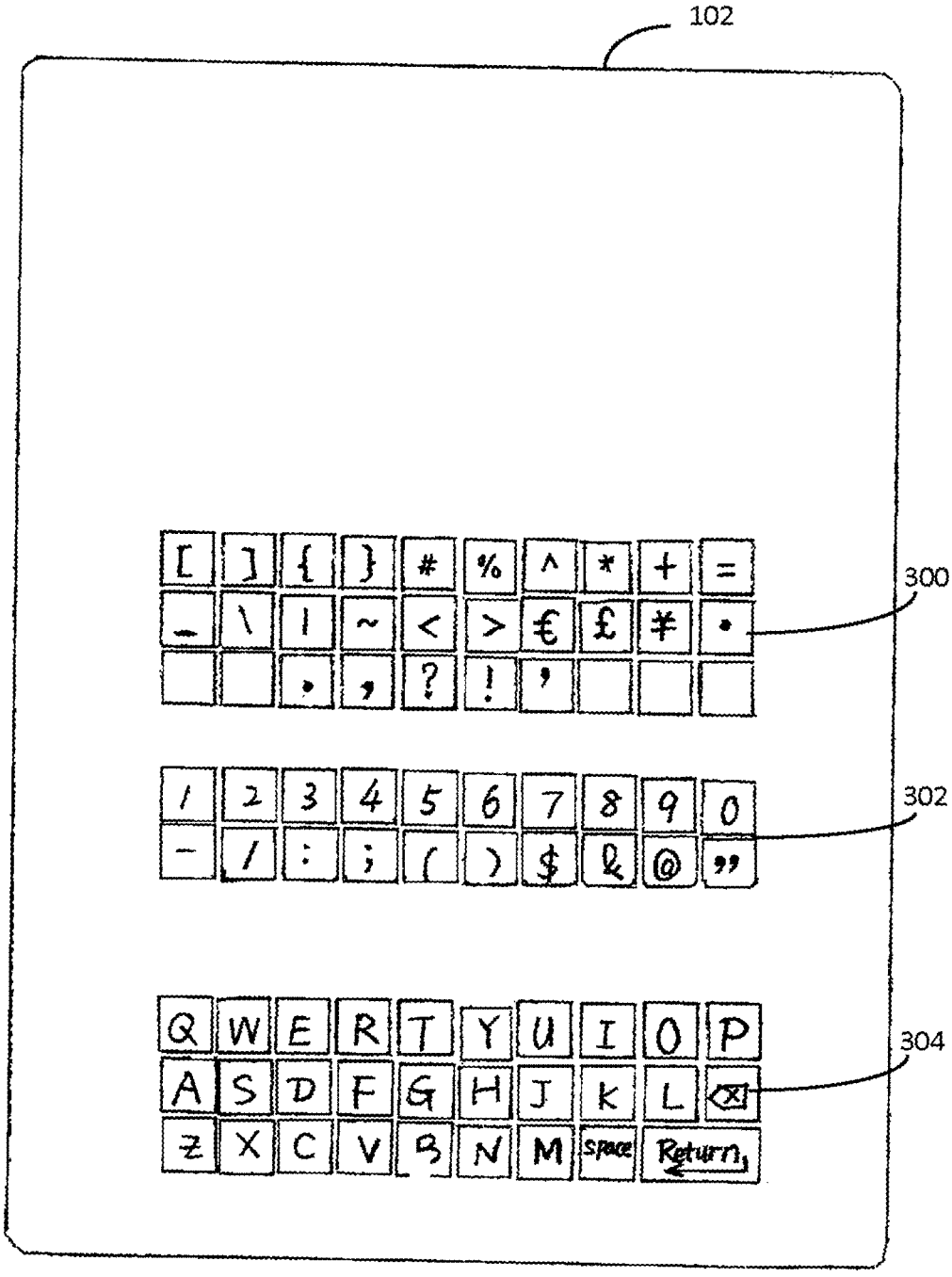


Figure 4

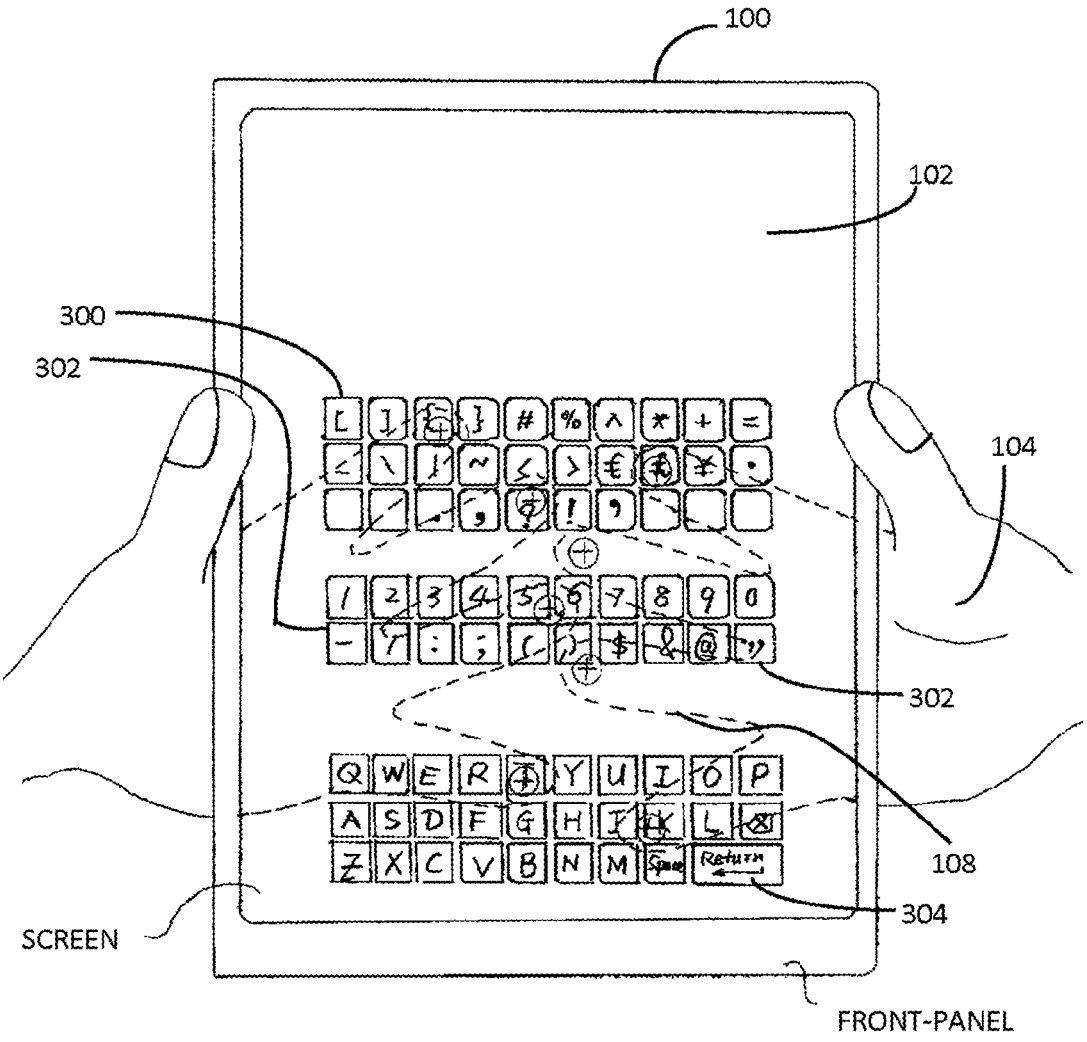


Figure 5

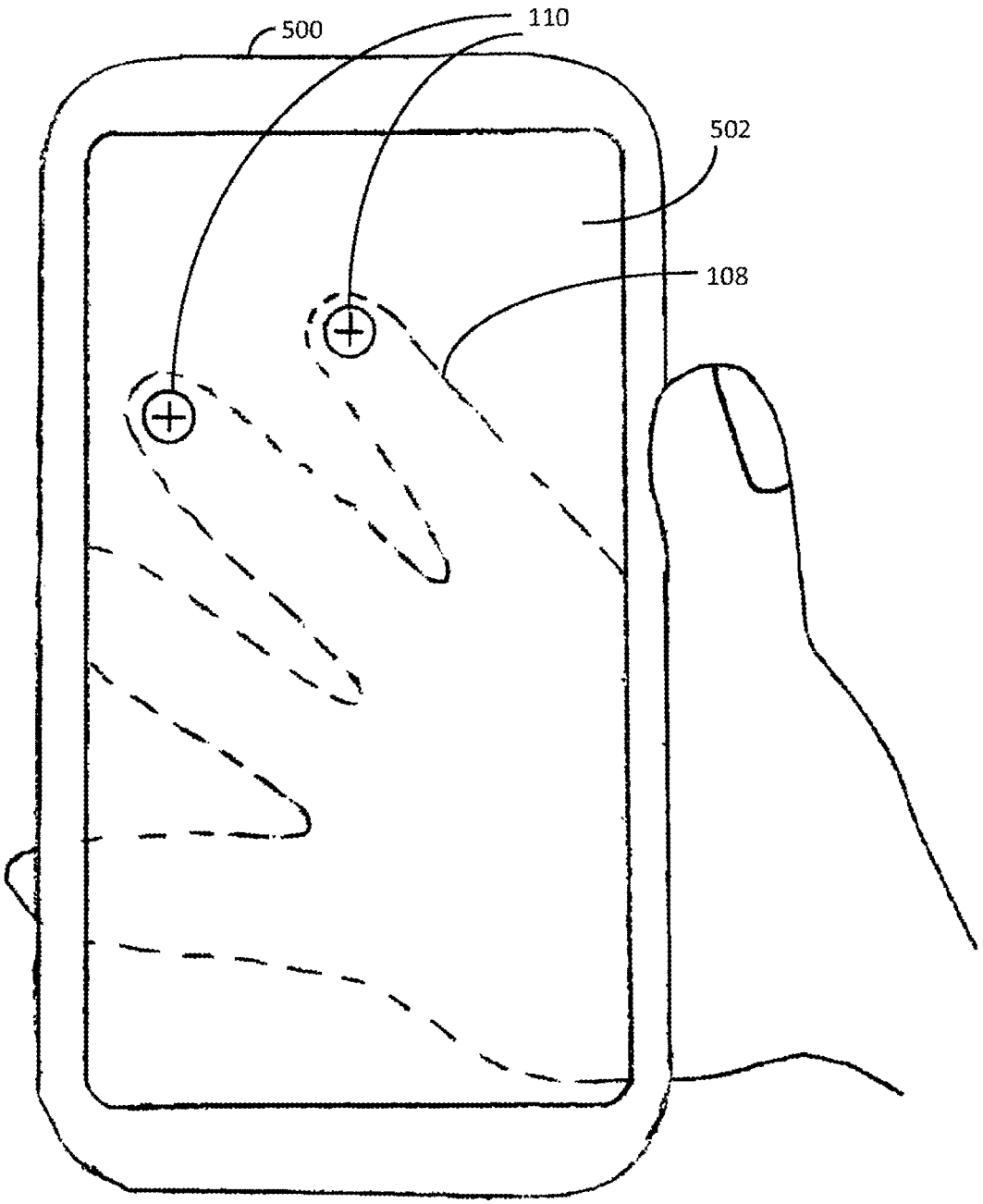


Figure 6

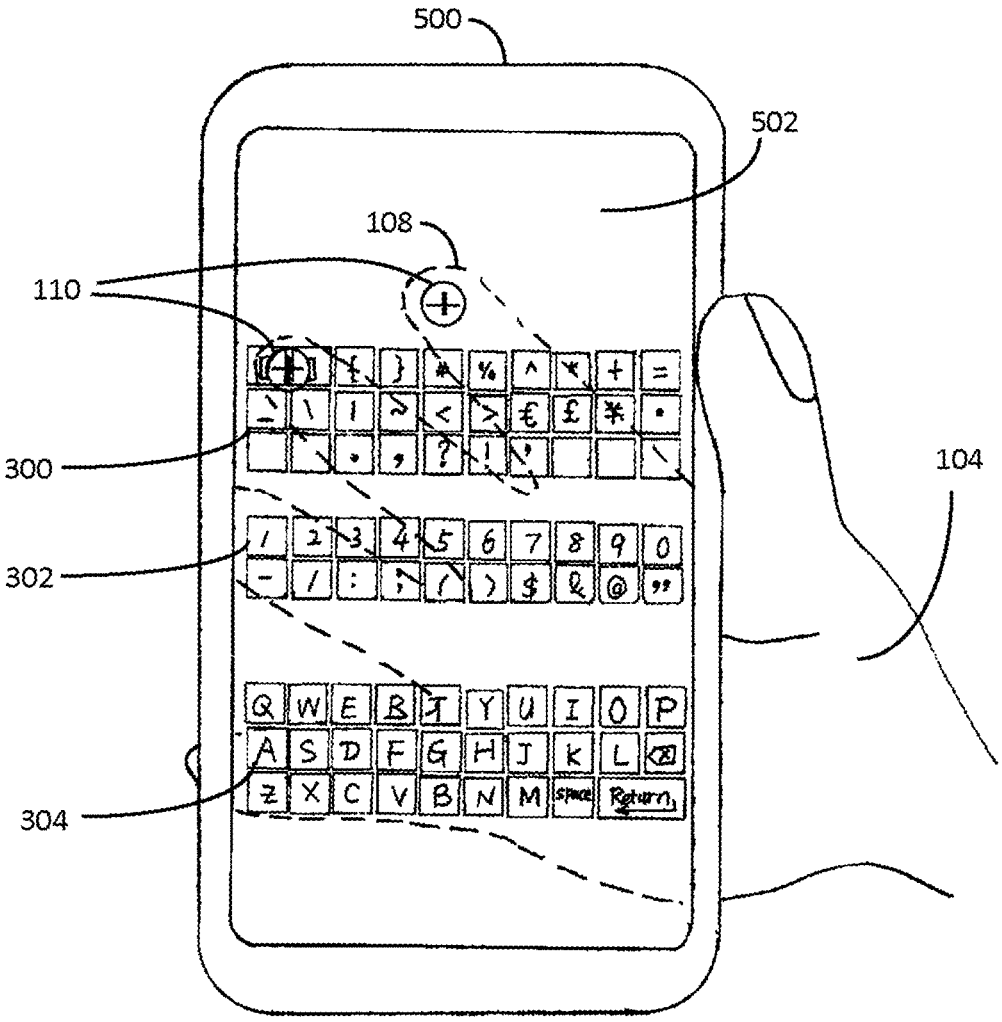


Figure 7

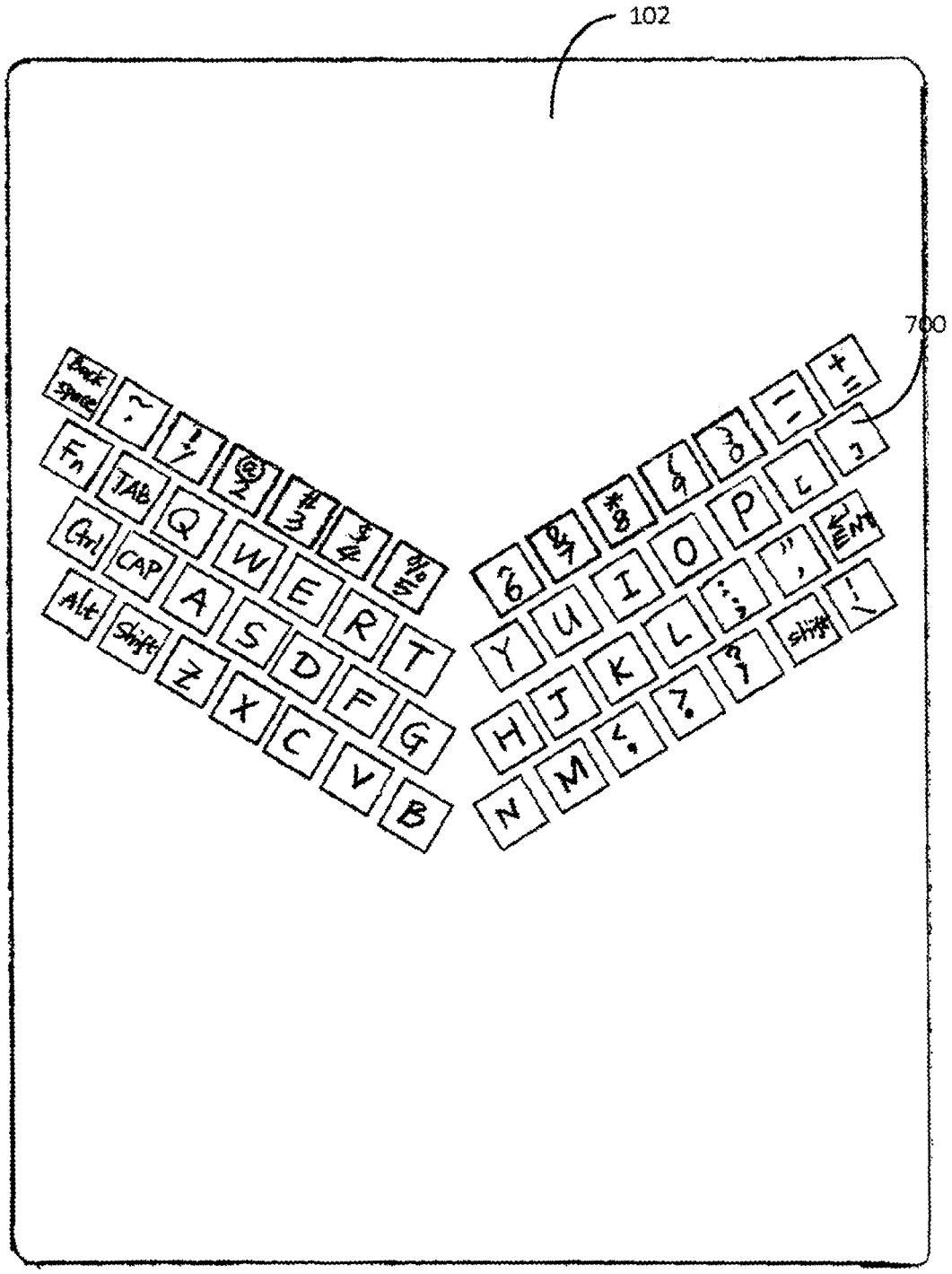


Figure 8

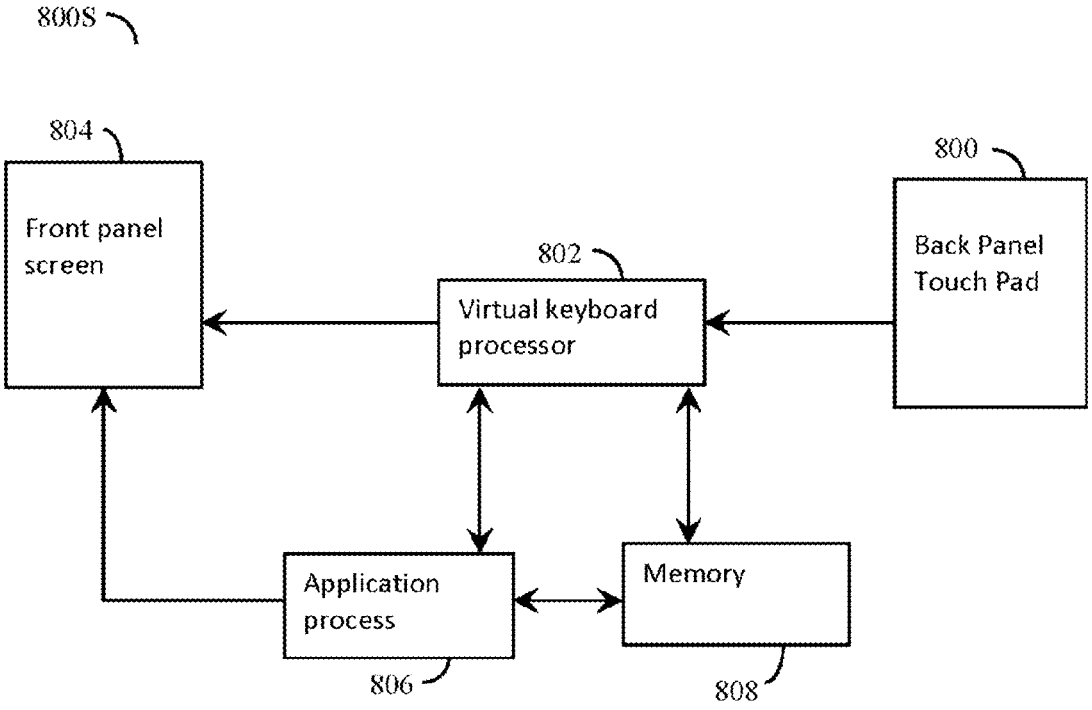

800S
804
Front panel screen
800
Back Panel Touch Pad
802
Virtual keyboard processor
Application process
806
Memory
808

Figure 9

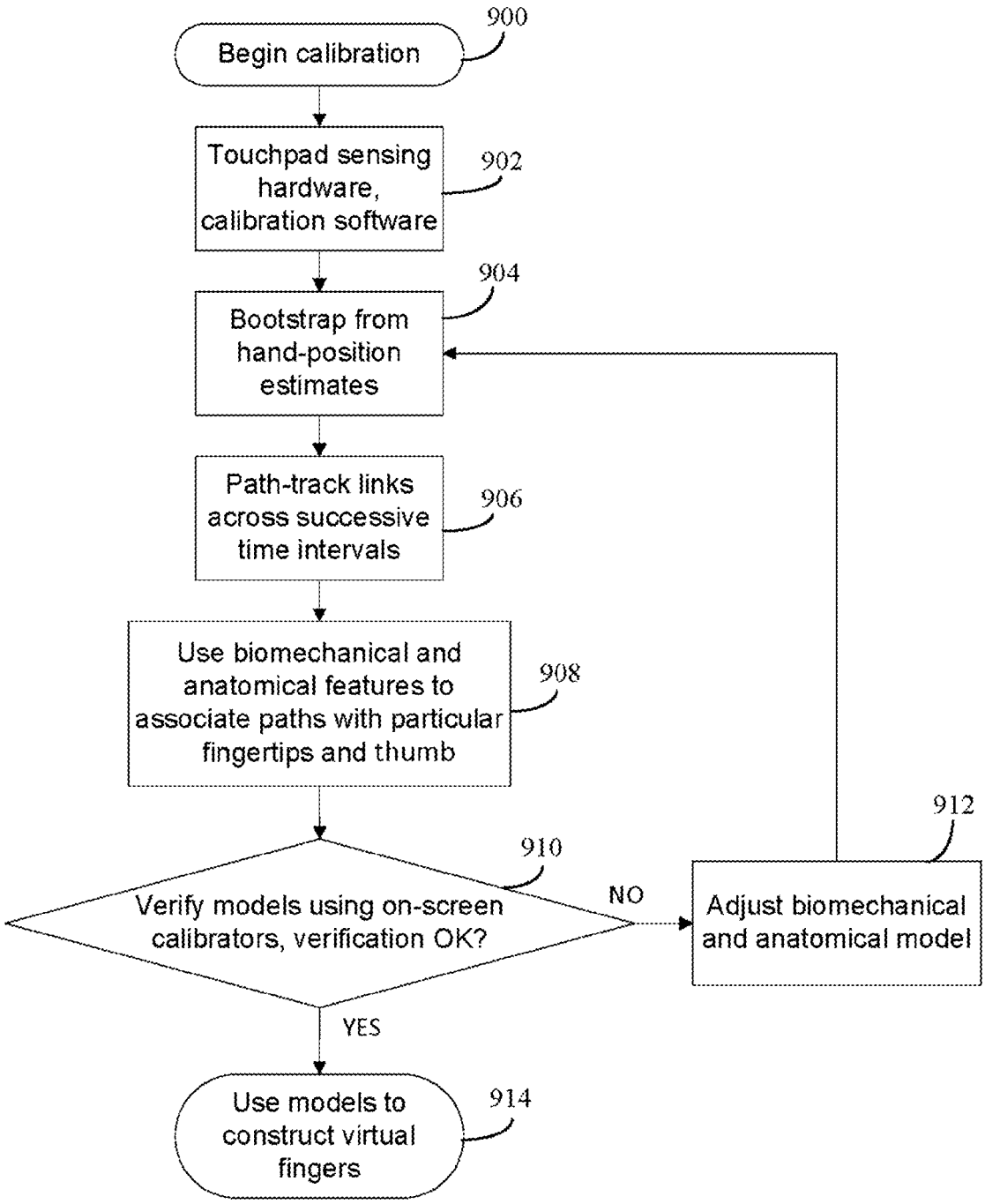

Begin calibration
900
Touchpad sensing hardware, calibration software
902
Bootstrap from hand-position estimates
904
Path-track links across successive time intervals
906
Use biomechanical and anatomical features to associate paths with particular fingertips and thumb
908
Verify models using on-screen calibrators, verification OK?
910
NO
Adjust biomechanical and anatomical model
912
YES
Use models to construct virtual fingers
914

Figure 10

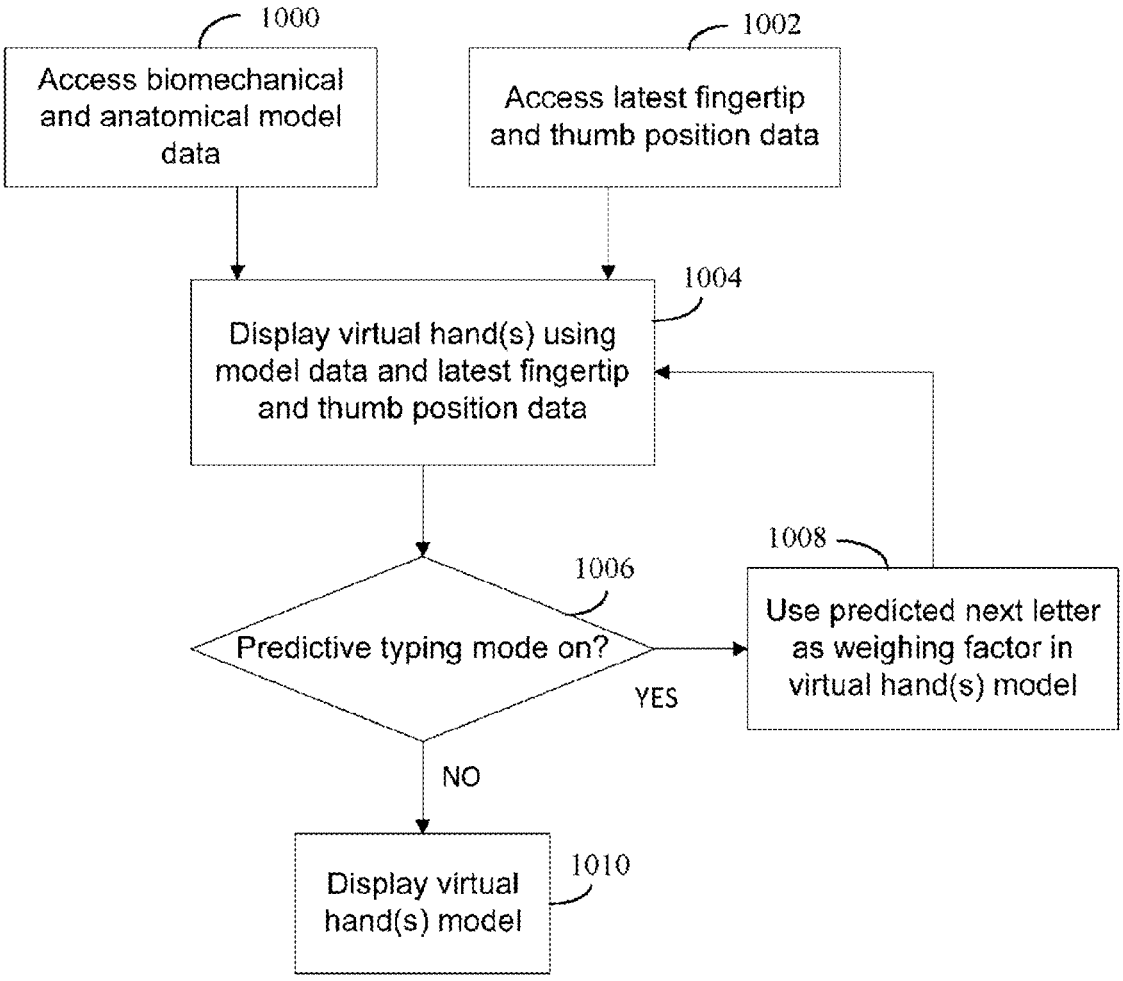

1000
Access biomechanical and anatomical model data
1002
Access latest fingertip and thumb position data
1004
Display virtual hand(s) using model data and latest fingertip and thumb position data
1006
Predictive typing mode on?
YES
NO
1008
Use predicted next letter as weighing factor in virtual hand(s) model
Display virtual hand(s) model
1010

Figure 11

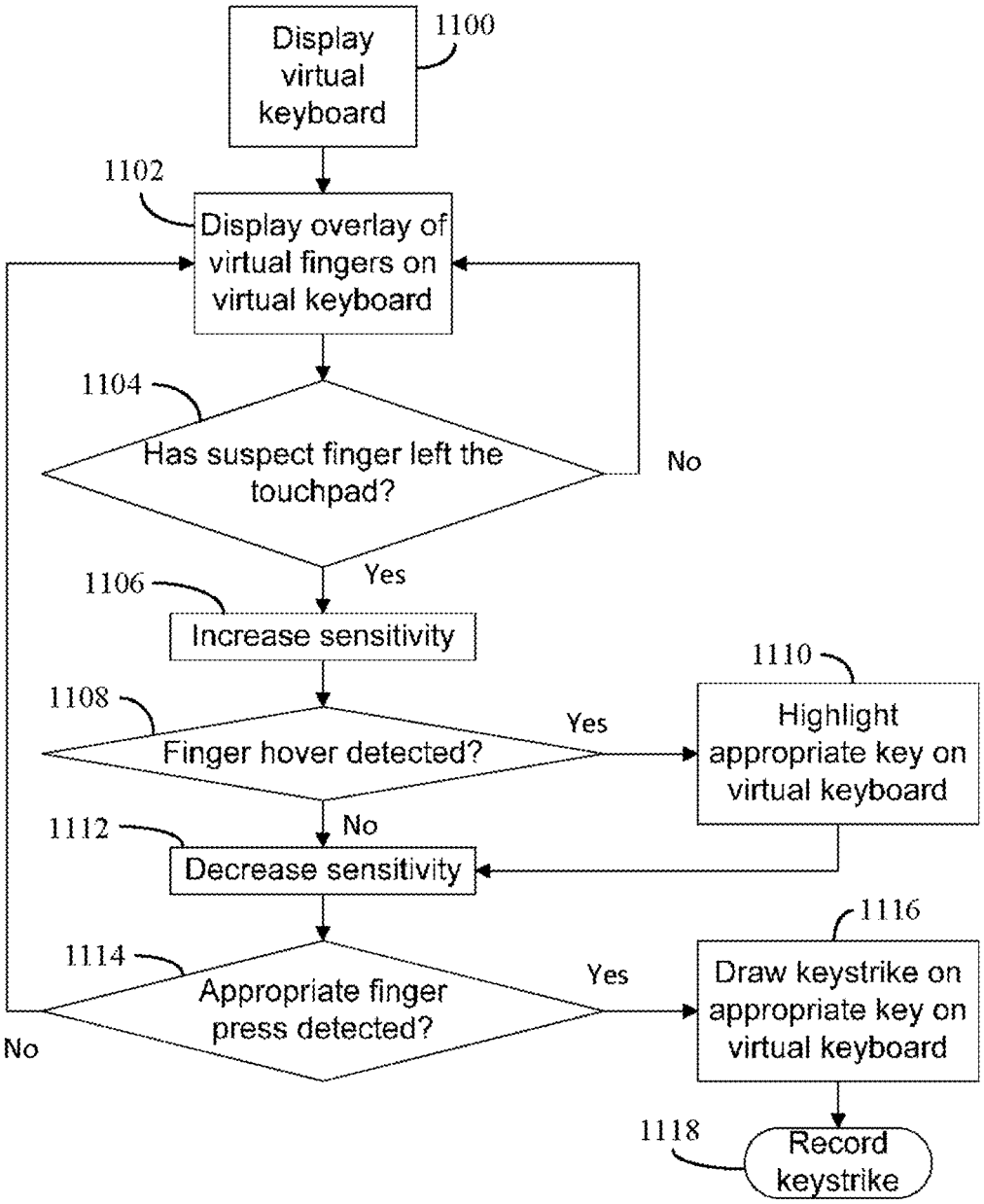

Display virtual keyboard
1100
1102
Display overlay of virtual fingers on virtual keyboard
1104
Has suspect finger left the touchpad?
No
Yes
1106
Increase sensitivity
1108
Finger hover detected?
Yes
1110
Highlight appropriate key on virtual keyboard
No
1112
Decrease sensitivity
1114
Appropriate finger press detected?
Yes
No
1116
Draw keystrike on appropriate key on virtual keyboard
1118
Record keystrike

Figure 12

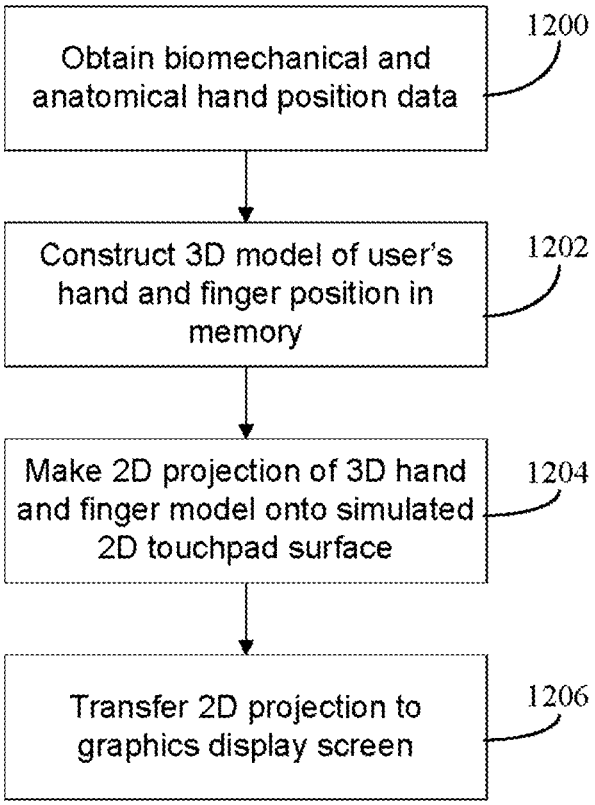

Obtain biomechanical and anatomical hand position data
1200
Construct 3D model of user's hand and finger position in memory
1202
Make 2D projection of 3D hand and finger model onto simulated 2D touchpad surface
1204
Transfer 2D projection to graphics display screen
1206

Figure 13

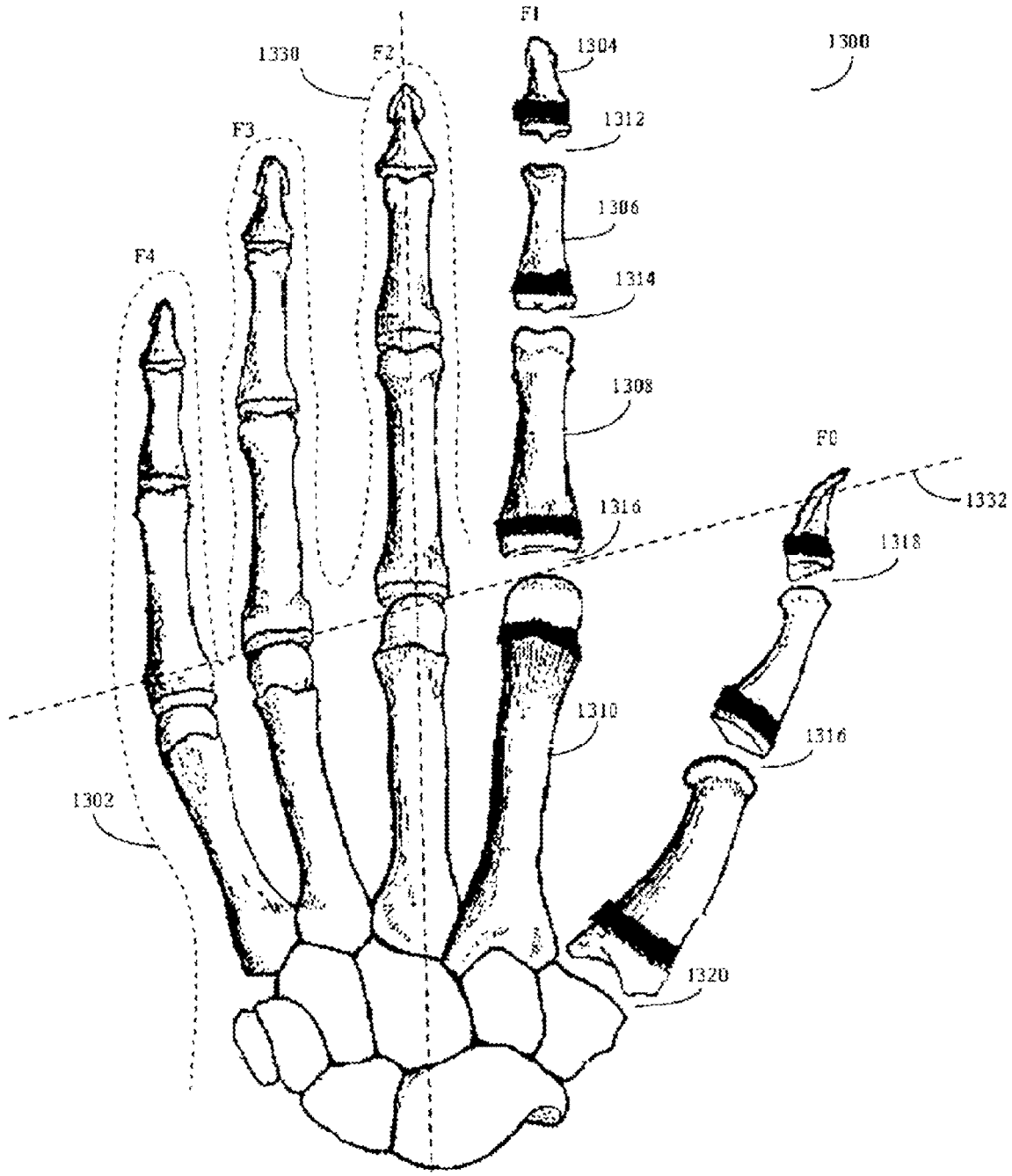
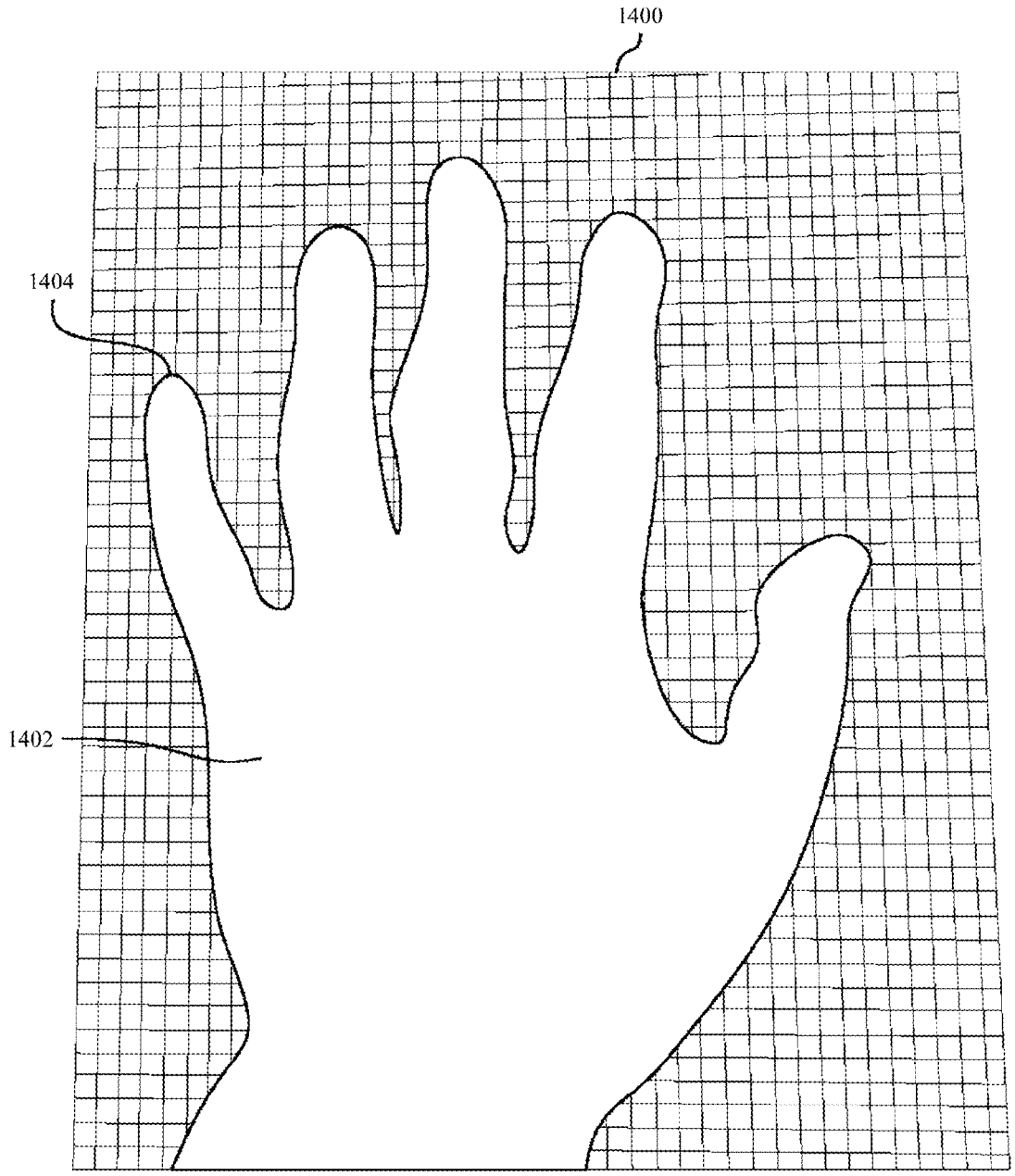

F1
1330
F2
1304
1300
F3
1312
1306
F4
1314
1308
F0
1332
1316
1318
1310
1316
1302
1320

Figure 14



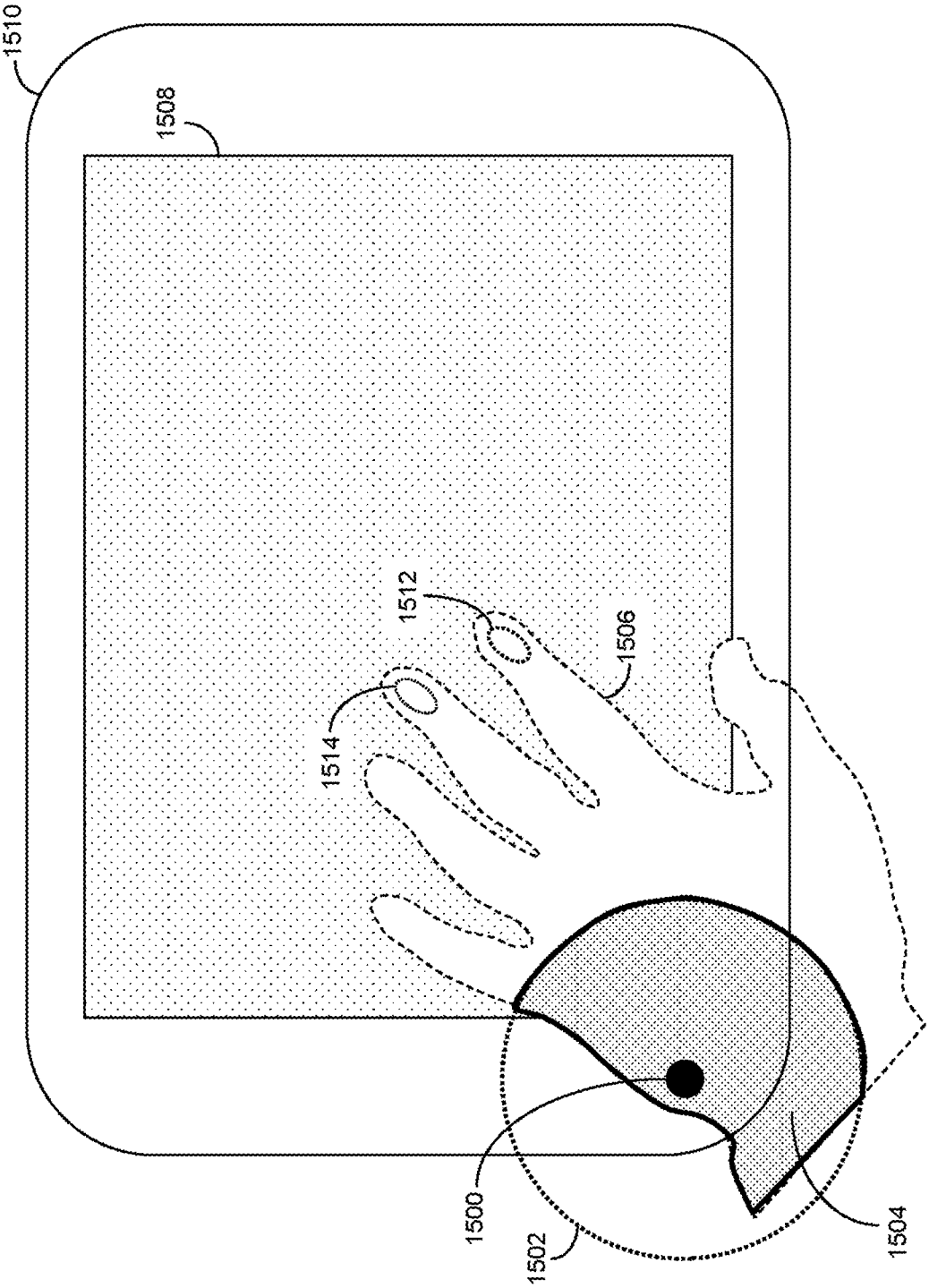


Figure 15

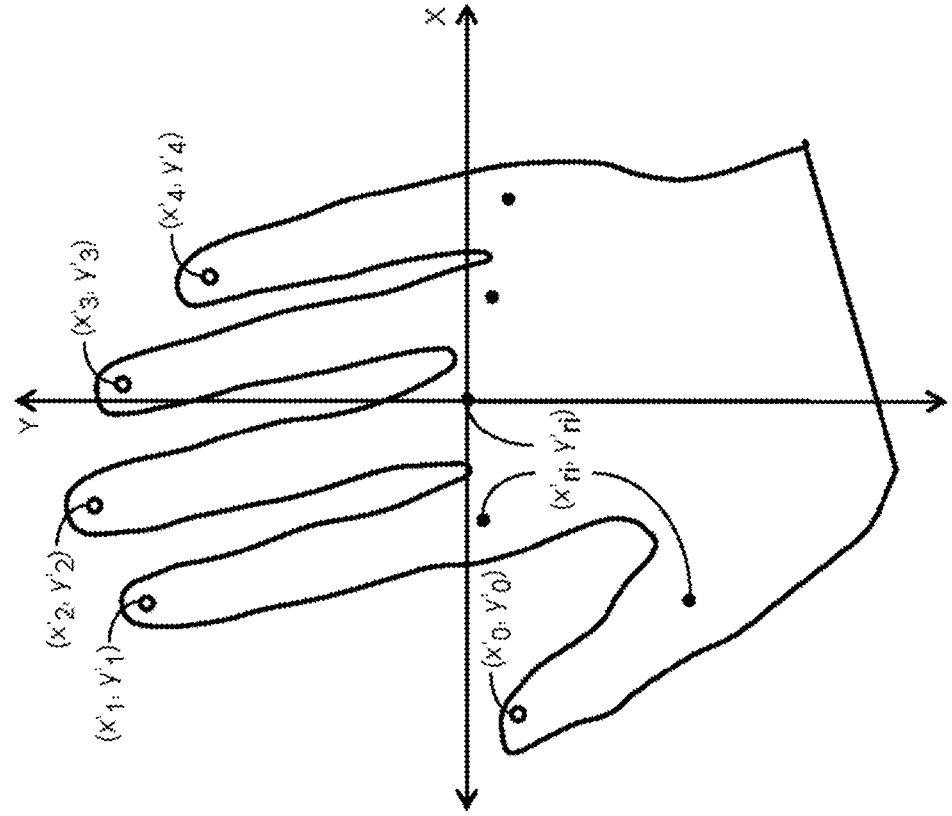


Figure 16B

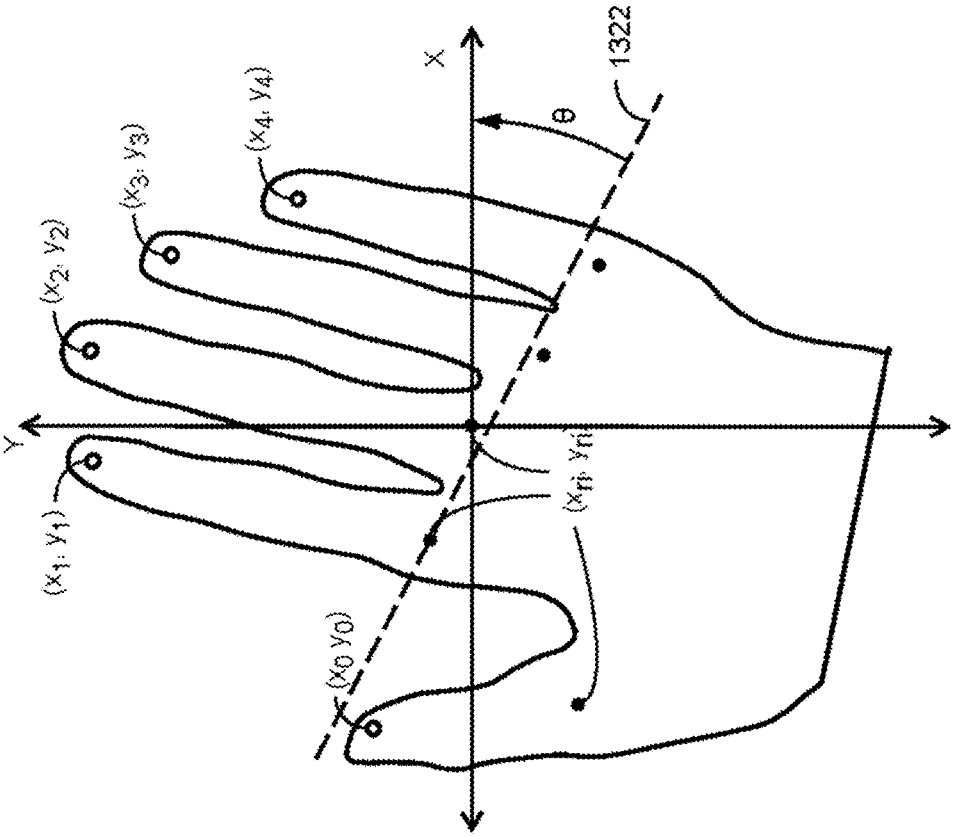


Figure 16A

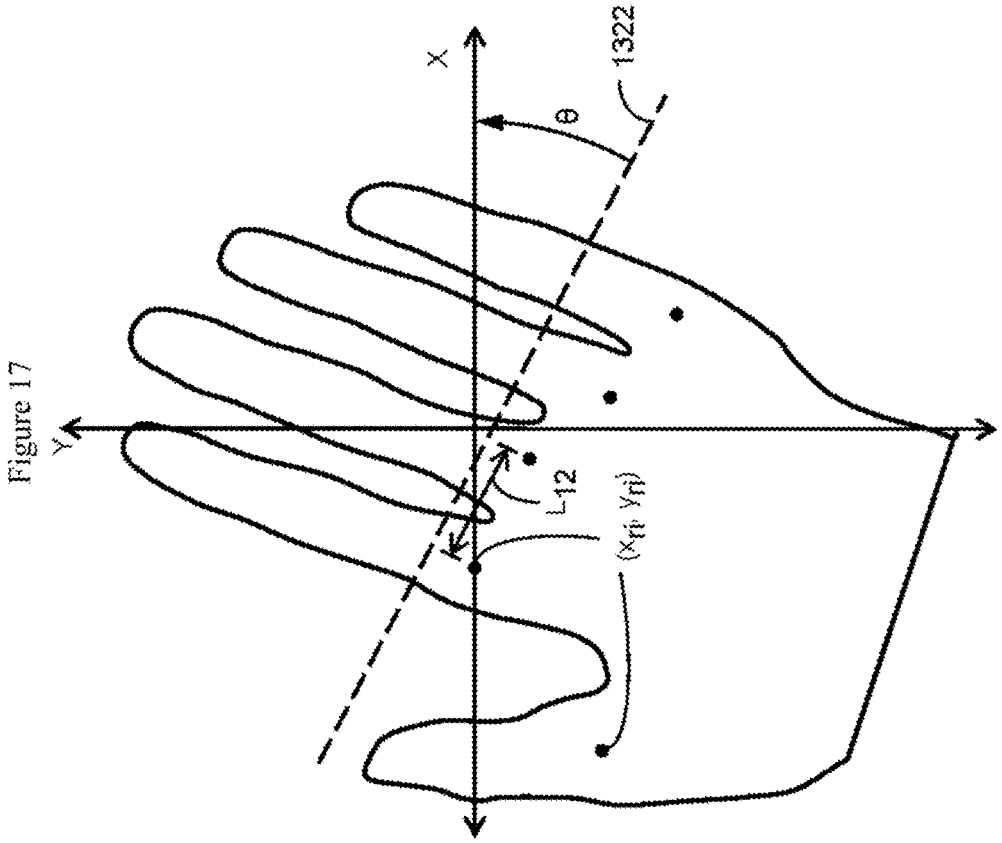
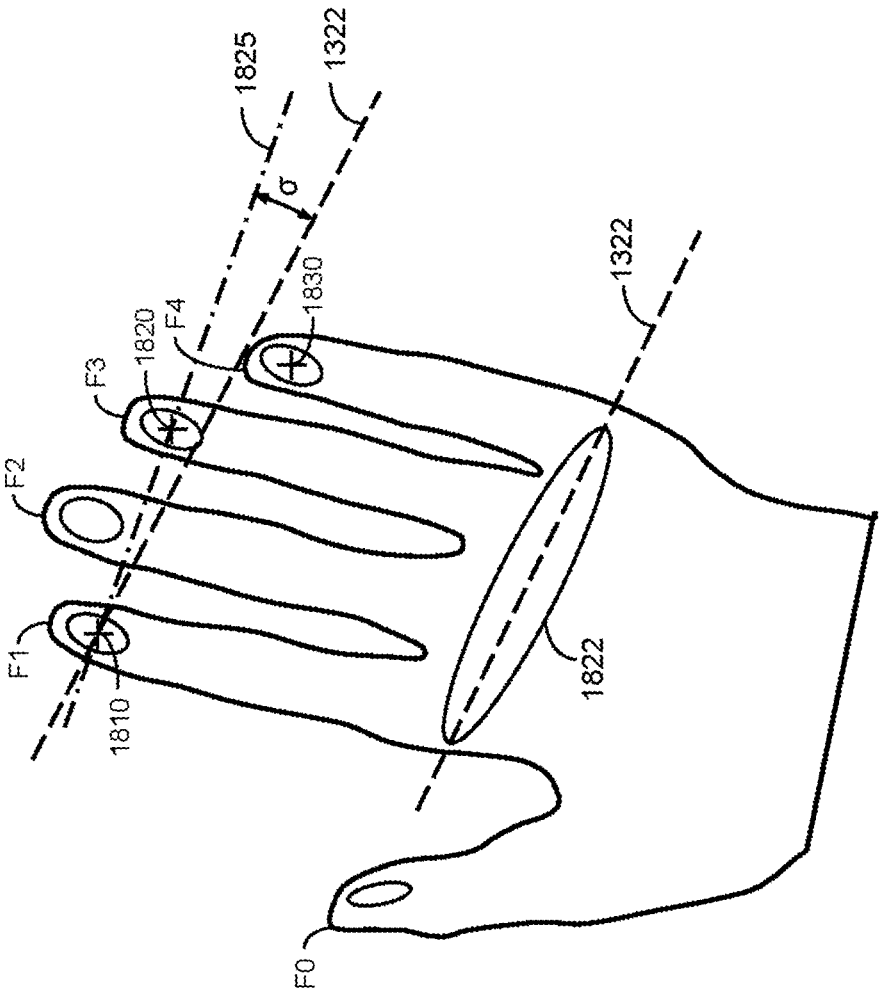


Figure 17

Figure 18



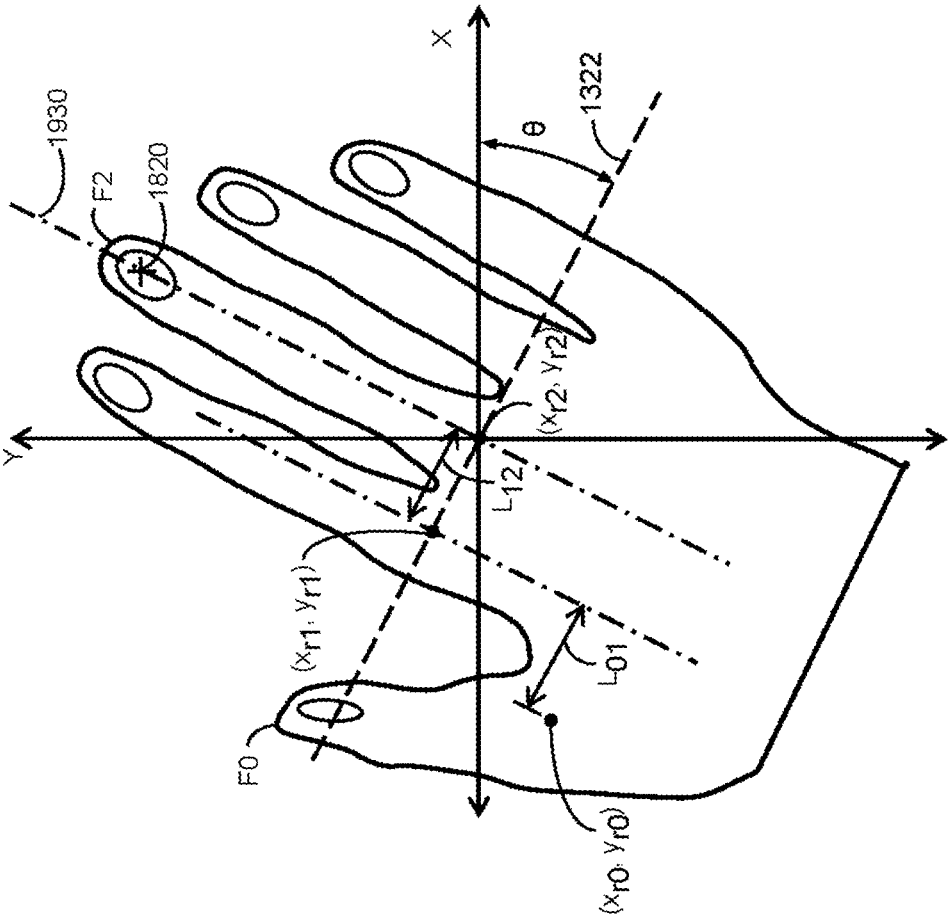


Figure 19

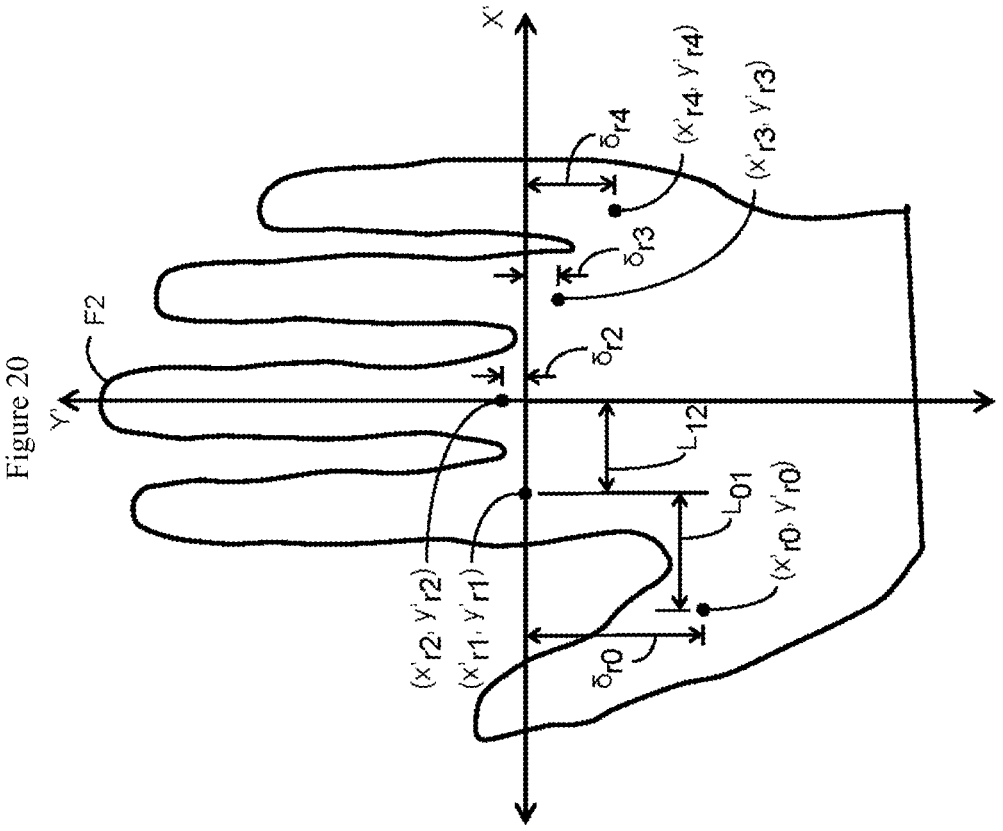

X'
Y'
F2
δ_{r4}
(x'_{r4}, y'_{r4})
(x'_{r3}, y'_{r3})
δ_{r3}
δ_{r2}
L_{12}
L_{01}
(x'_{r0}, y'_{r0})
(x'_{r2}, y'_{r2})
(x'_{r1}, y'_{r1})
δ_{r0}

Figure 20

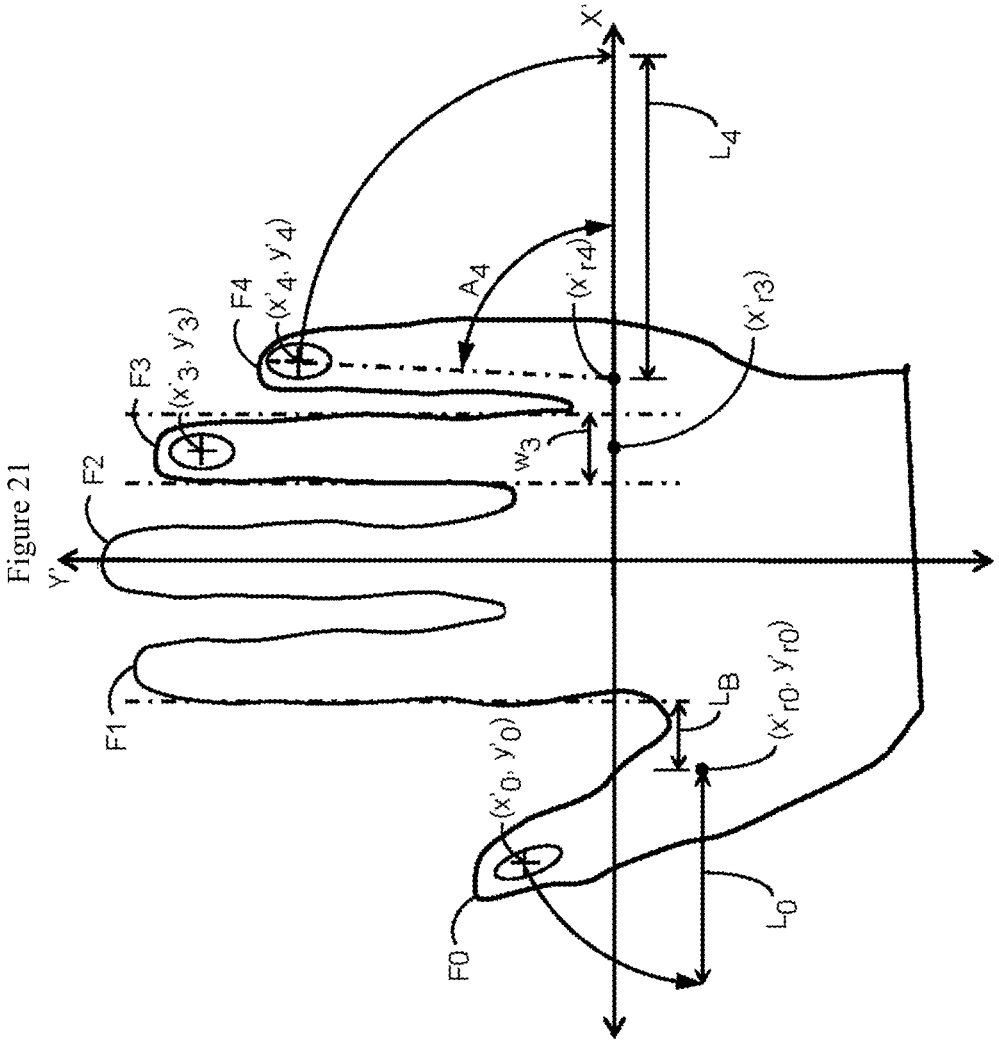


Figure 21

Figure 22

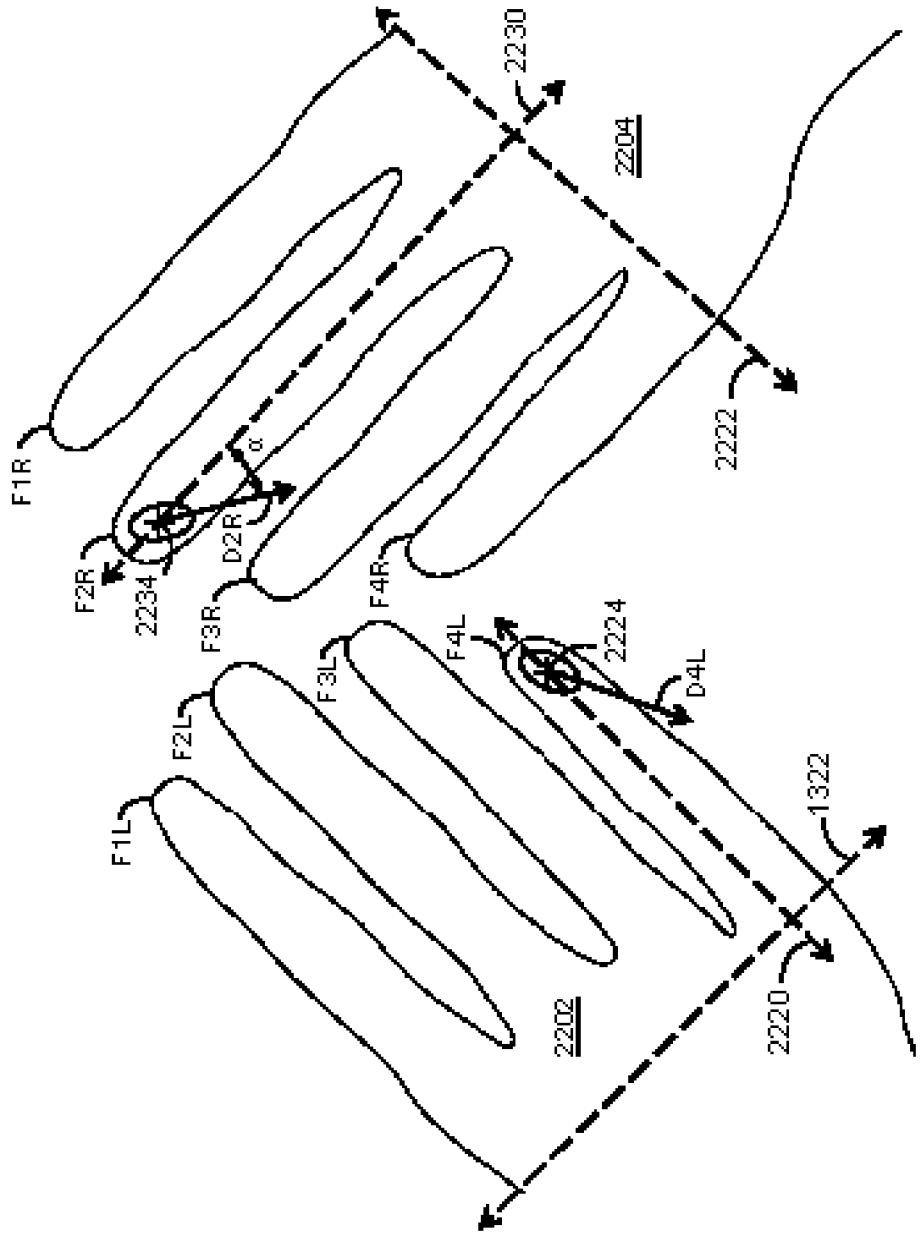
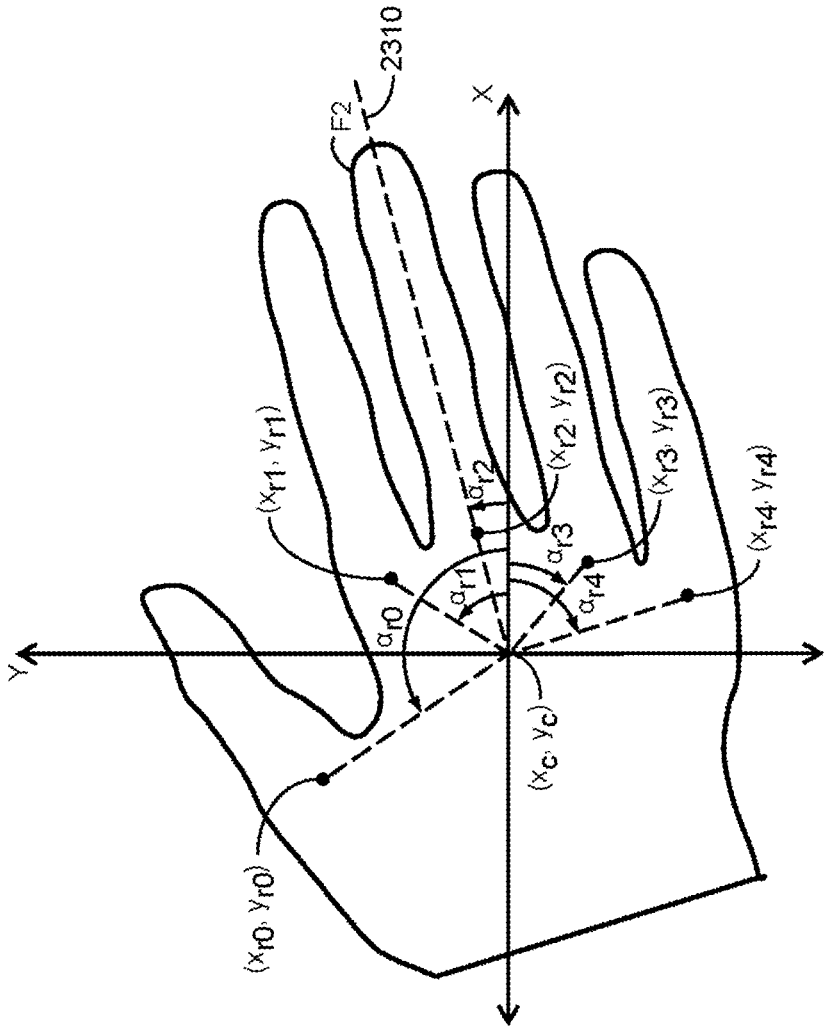

2230
2204
2222
F1R
F2R
2234
D2R
α
F3R
F4R
F4L
2224
D4L
F3L
F2L
F1L
1322
2220
2202

Figure 23


2310
F2
X
Y
(x_{r1}, y_{r1})
(x_{r2}, y_{r2})
(x_{r3}, y_{r3})
(x_{r4}, y_{r4})
(x_{r0}, y_{r0})
(x_c, y_c)
α_{r0}
α_{r1}
α_{r2}
α_{r3}
α_{r4}

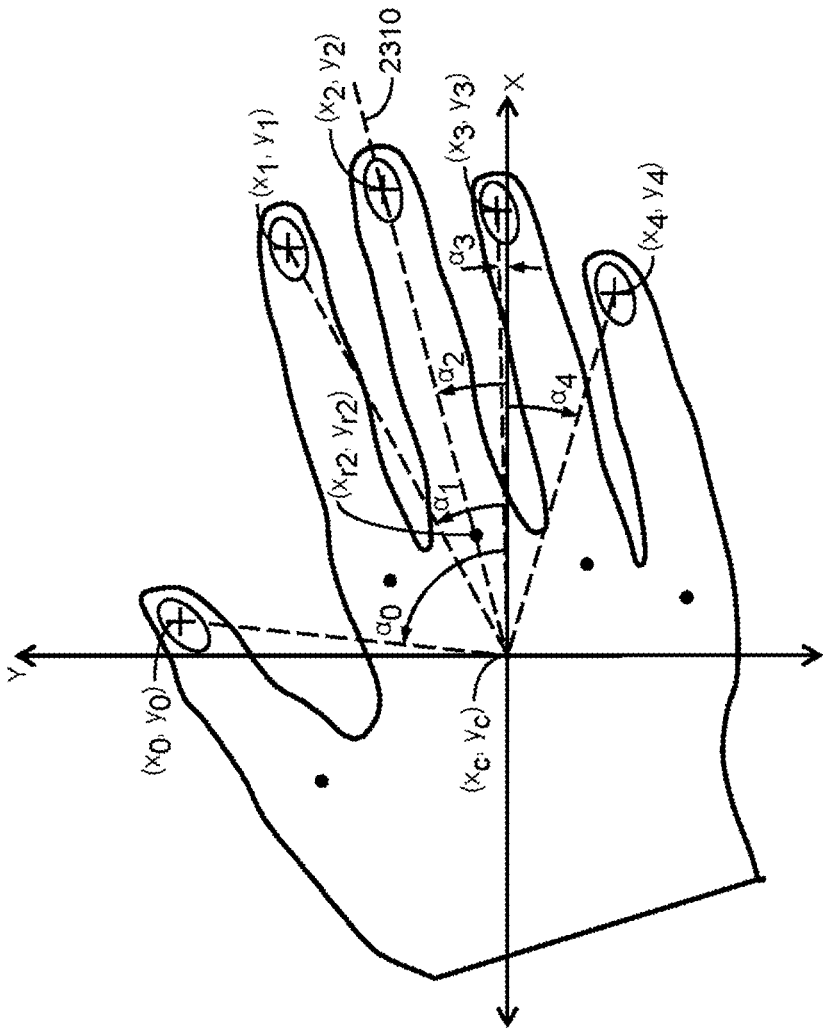

2310
X
Y
(x_1, y_1)
(x_2, y_2)
(x_3, y_3)
(x_4, y_4)
(x_{r2}, y_{r2})
(x_0, y_0)
(x_c, y_c)
α_0
α_1
α_2
α_3
α_4

Figure 24

Figure 25

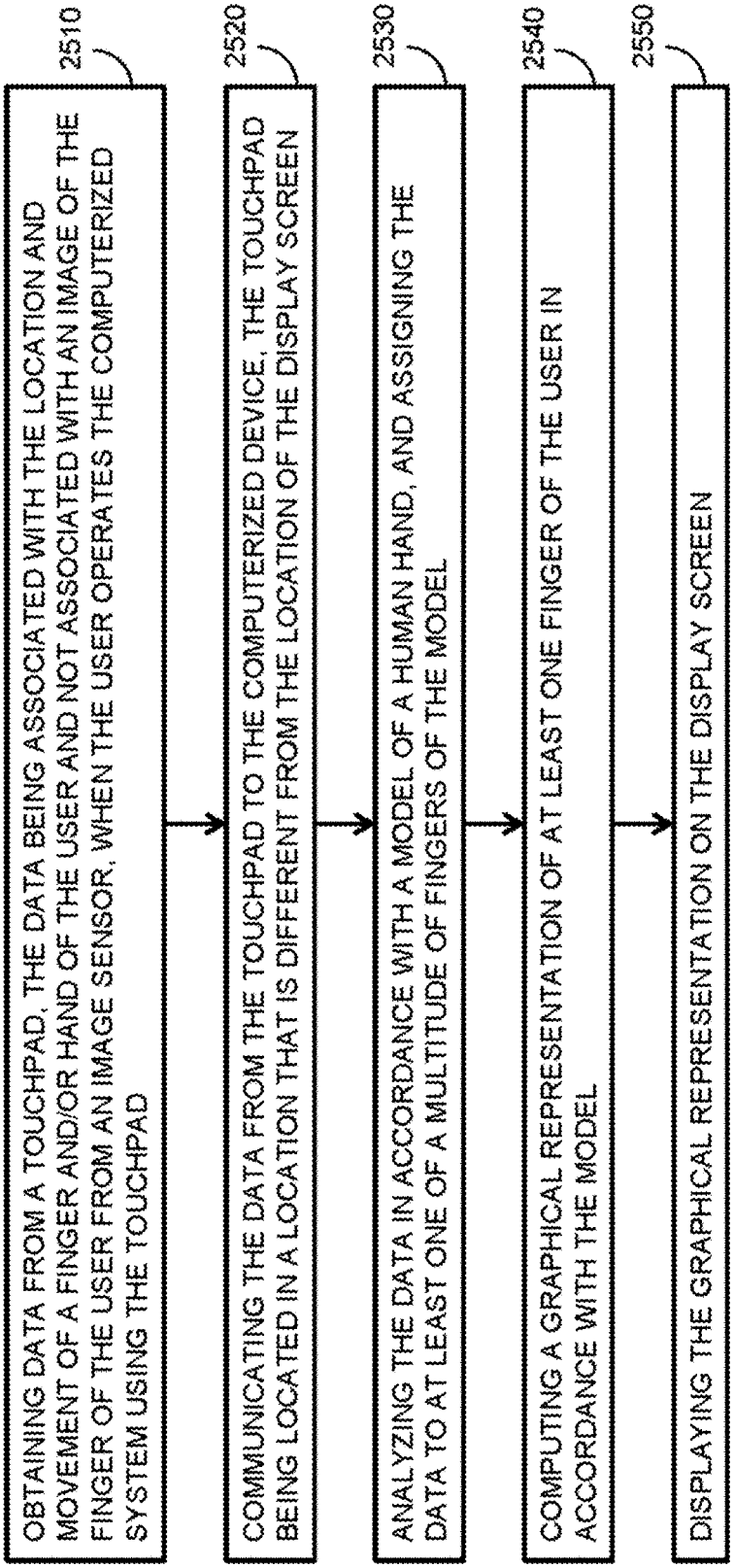
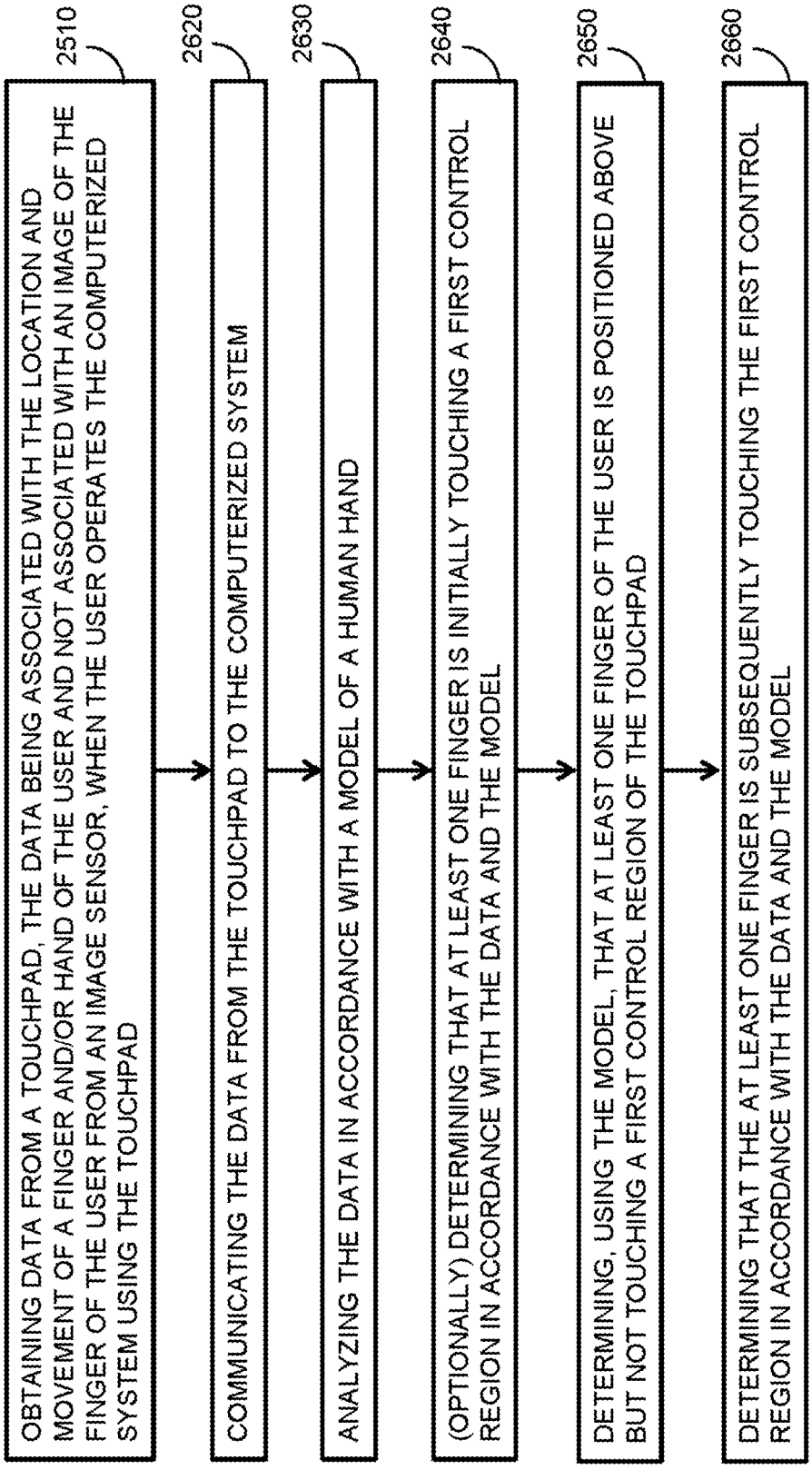

2510
OBTAINING DATA FROM A TOUCHPAD, THE DATA BEING ASSOCIATED WITH THE LOCATION AND MOVEMENT OF A FINGER AND/OR HAND OF THE USER AND NOT ASSOCIATED WITH AN IMAGE OF THE FINGER OF THE USER FROM AN IMAGE SENSOR, WHEN THE USER OPERATES THE COMPUTERIZED SYSTEM USING THE TOUCHPAD
2520
COMMUNICATING THE DATA FROM THE TOUCHPAD TO THE COMPUTERIZED DEVICE, THE TOUCHPAD BEING LOCATED IN A LOCATION THAT IS DIFFERENT FROM THE LOCATION OF THE DISPLAY SCREEN
2530
ANALYZING THE DATA IN ACCORDANCE WITH A MODEL OF A HUMAN HAND, AND ASSIGNING THE DATA TO AT LEAST ONE OF A MULTITUDE OF FINGERS OF THE MODEL
2540
COMPUTING A GRAPHICAL REPRESENTATION OF AT LEAST ONE FINGER OF THE USER IN ACCORDANCE WITH THE MODEL
2550
DISPLAYING THE GRAPHICAL REPRESENTATION ON THE DISPLAY SCREEN

Figure 26


2510
OBTAINING DATA FROM A TOUCHPAD, THE DATA BEING ASSOCIATED WITH THE LOCATION AND MOVEMENT OF A FINGER AND/OR HAND OF THE USER AND NOT ASSOCIATED WITH AN IMAGE OF THE FINGER OF THE USER FROM AN IMAGE SENSOR, WHEN THE USER OPERATES THE COMPUTERIZED SYSTEM USING THE TOUCHPAD
2620
COMMUNICATING THE DATA FROM THE TOUCHPAD TO THE COMPUTERIZED SYSTEM
2630
ANALYZING THE DATA IN ACCORDANCE WITH A MODEL OF A HUMAN HAND
2640
(OPTIONALLY) DETERMINING THAT AT LEAST ONE FINGER IS INITIALLY TOUCHING A FIRST CONTROL REGION IN ACCORDANCE WITH THE DATA AND THE MODEL
2650
DETERMINING, USING THE MODEL, THAT AT LEAST ONE FINGER OF THE USER IS POSITIONED ABOVE BUT NOT TOUCHING A FIRST CONTROL REGION OF THE TOUCHPAD
2660
DETERMINING THAT THE AT LEAST ONE FINGER IS SUBSEQUENTLY TOUCHING THE FIRST CONTROL REGION IN ACCORDANCE WITH THE DATA AND THE MODEL

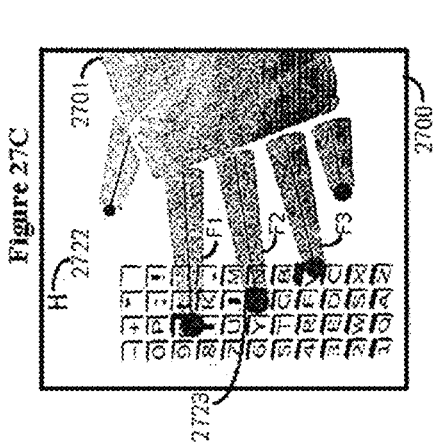


Figure 27C
2701
2700
H
2722
F1
F2
F3
2723

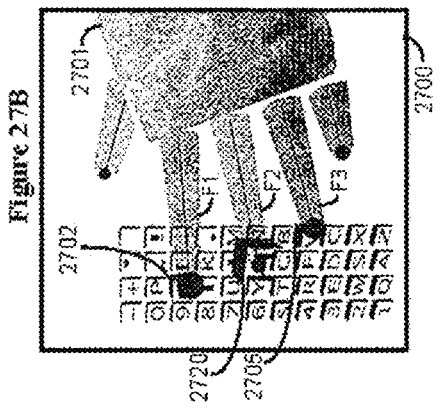


Figure 27B
2701
2700
2702
F1
F2
F3
2720
2705

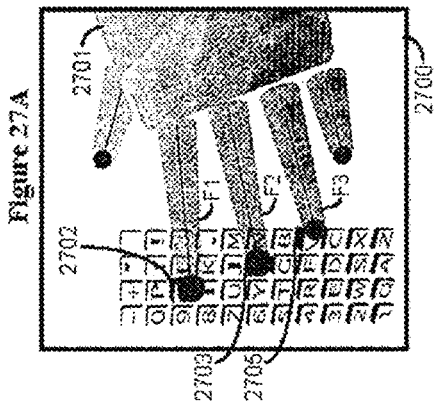


Figure 27A
2701
2700
2702
F1
F2
F3
2703
2705

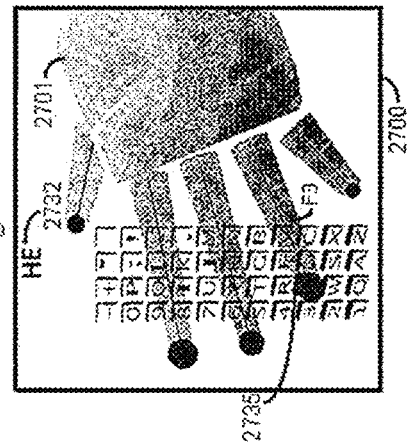


Figure 27F
2701
2700
HE
2732
F3
2735

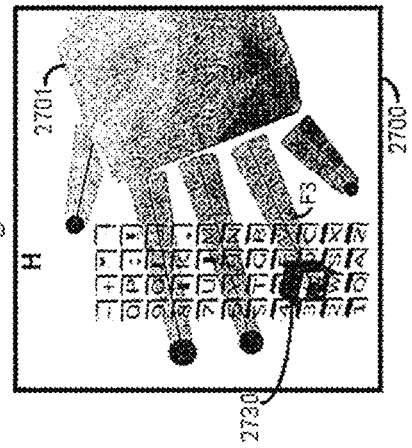


Figure 27E
2701
2700
H
F3
2730

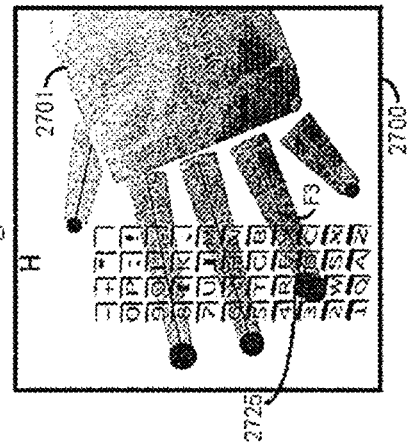


Figure 27D
2701
2700
H
F3
2725

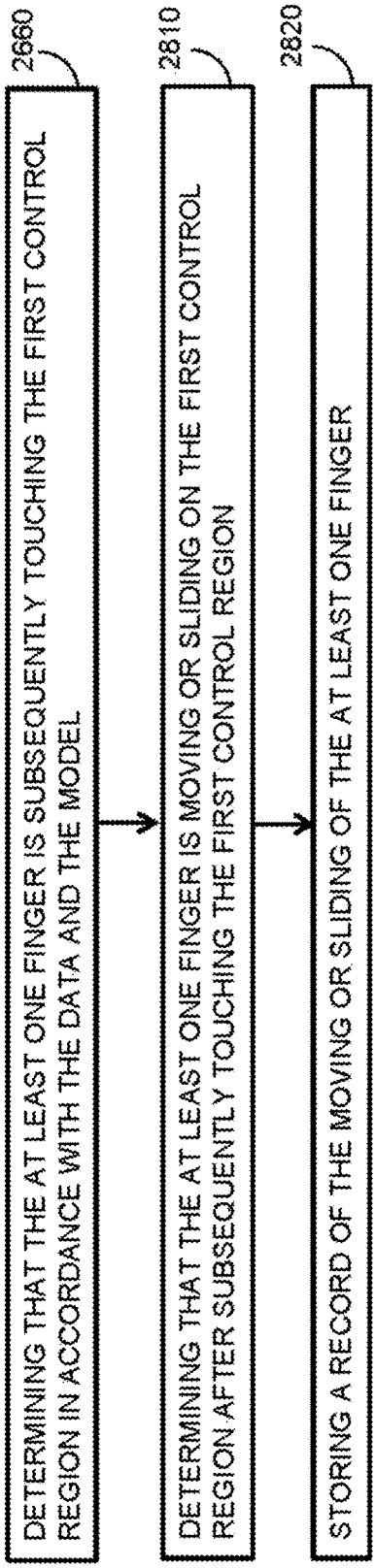

2660
DETERMINING THAT THE AT LEAST ONE FINGER IS SUBSEQUENTLY TOUCHING THE FIRST CONTROL REGION IN ACCORDANCE WITH THE DATA AND THE MODEL
2810
DETERMINING THAT THE AT LEAST ONE FINGER IS MOVING OR SLIDING ON THE FIRST CONTROL REGION AFTER SUBSEQUENTLY TOUCHING THE FIRST CONTROL REGION
2820
STORING A RECORD OF THE MOVING OR SLIDING OF THE AT LEAST ONE FINGER

Figure 28

Figure 29

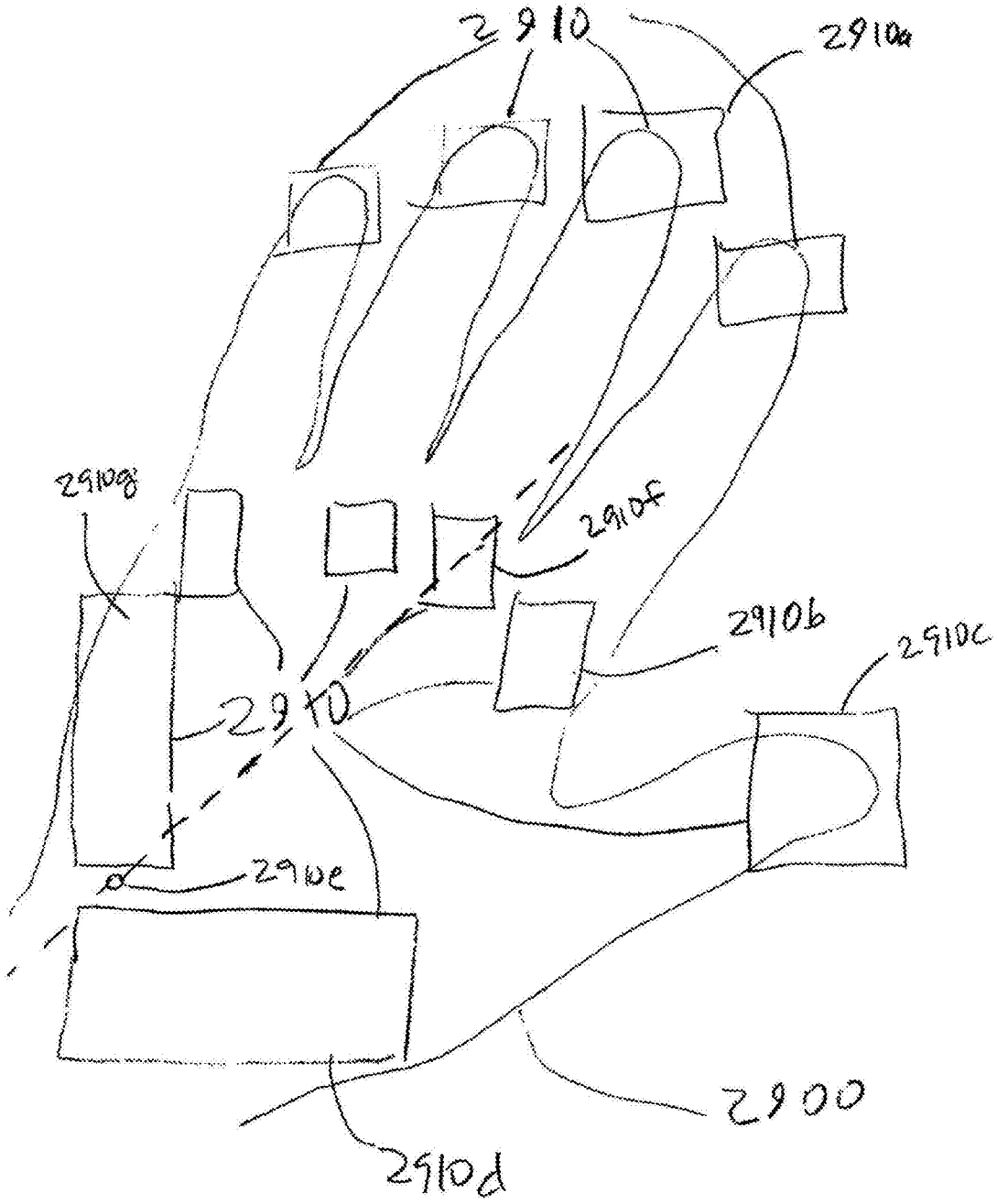


Figure 30

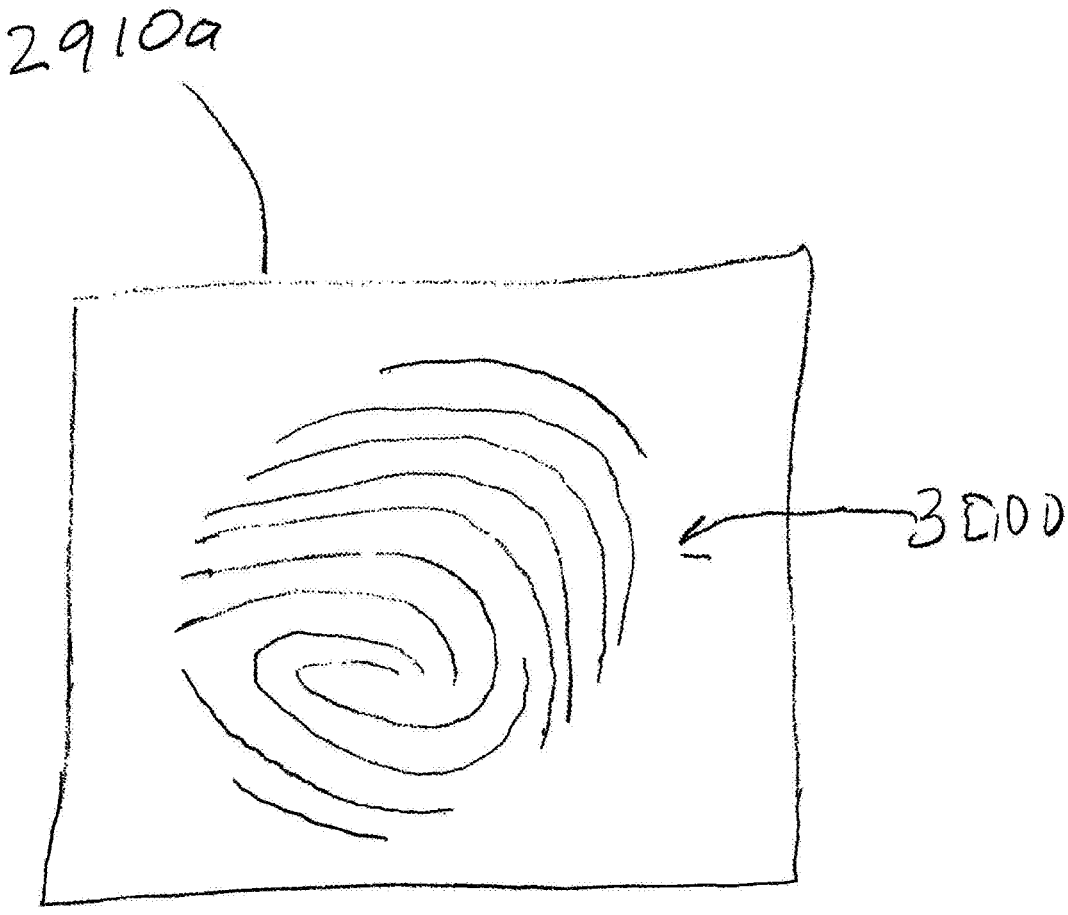


Figure 31

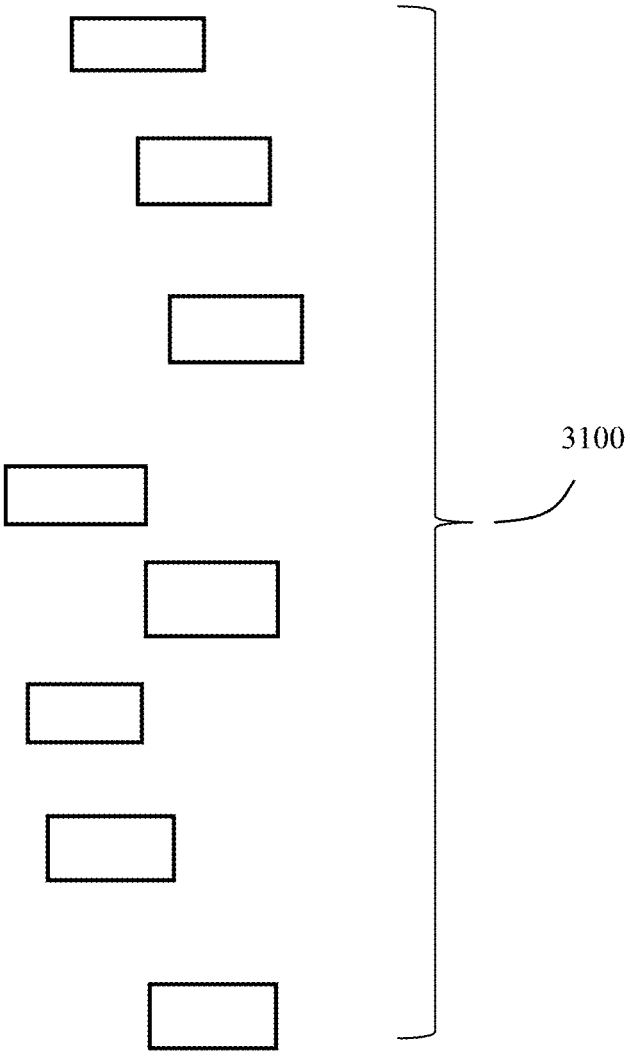


Figure 32

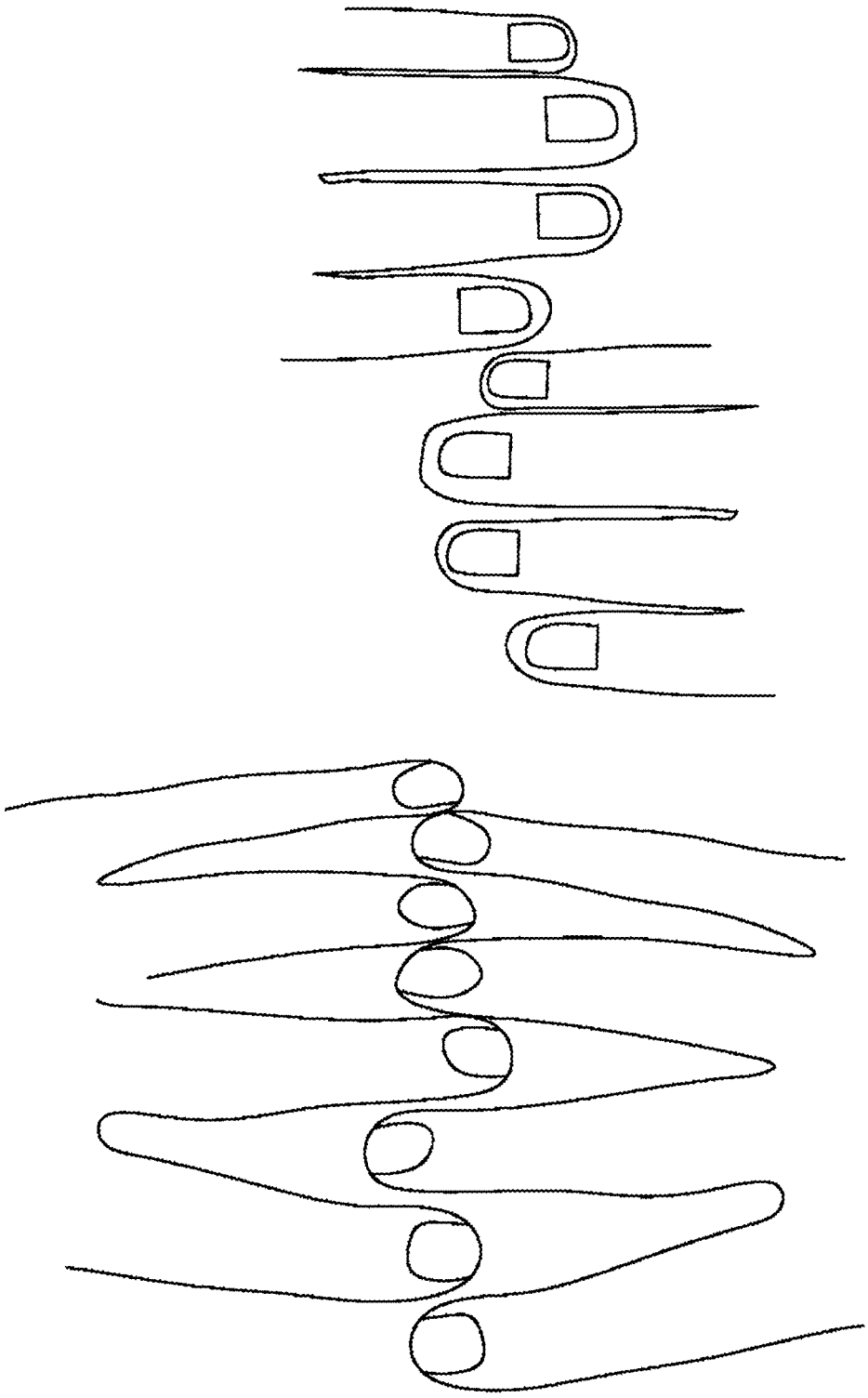


Figure 33

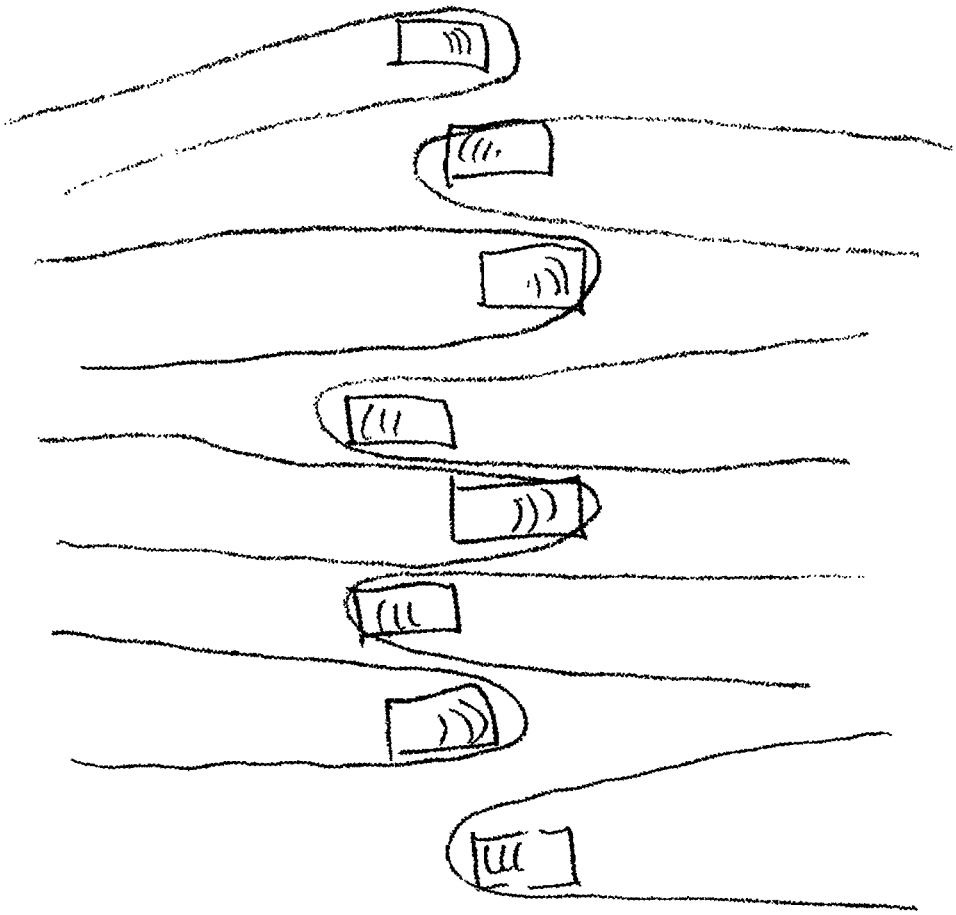
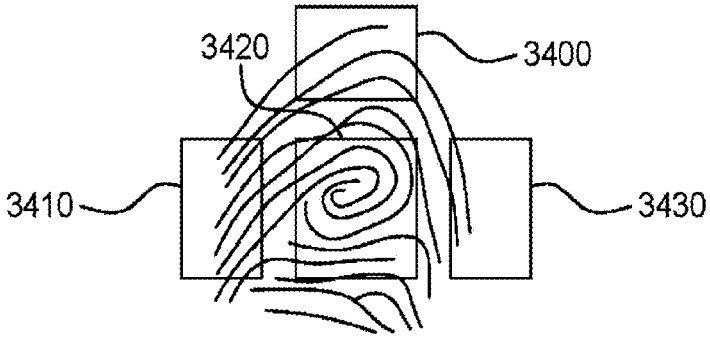
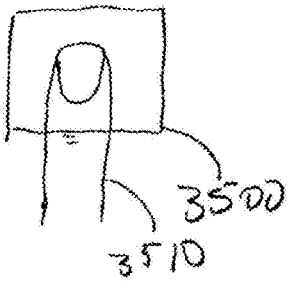


Figure 34





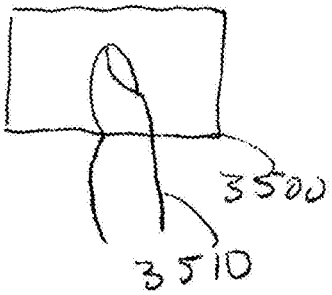


Figure 35 Figure 36 Figure 37

Figure 38

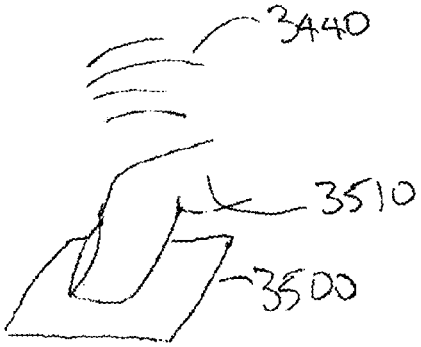


Figure 39

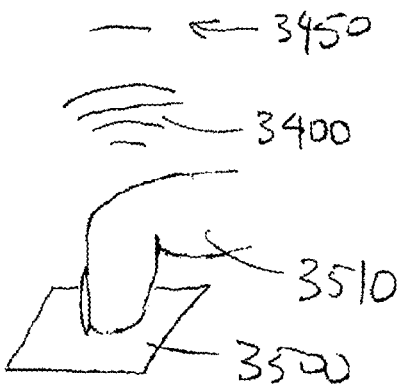
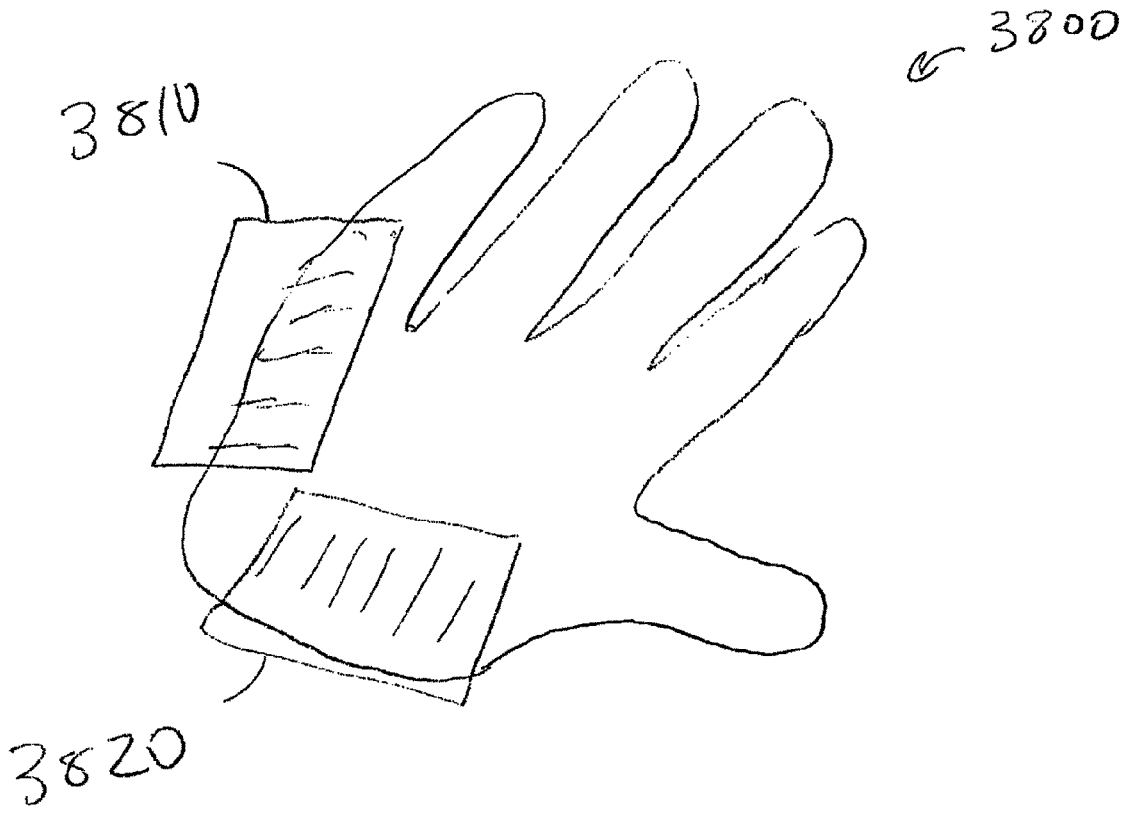


Figure 40



METHOD USING A TOUCHPAD FOR CONTROLLING A COMPUTERIZED SYSTEM WITH EPIDERMAL PRINT INFORMATION

CROSS-REFERENCES TO RELATED APPLICATIONS

This application claims the priority benefit of U.S. provisional patent application 62/016,026, entitled "METHOD USING A TOUCHPAD FOR CONTROLLING A COMPUTERIZED SYSTEM WITH EPIDERMAL PRINT INFORMATION", filed Jun. 23, 2014, and the priority benefit of U.S. provisional patent application 62/016,028 entitled, "METHOD FOR ORIENTING A HAND ON A TOUCHPAD OF A COMPUTERIZED SYSTEM", also filed Jun. 23, 2014; This application is also a continuation-in-part of U.S. patent application Ser. No. 14/268,926, entitled "METHOD FOR USER INPUT FROM ALTERNATIVE TOUCHPADS OF A HANDHELD COMPUTERIZED DEVICE," filed on May 2, 2014, which claims priority, under 35 USC §119(e), from U.S. Provisional Patent Application No. 61/819,615, entitled "METHOD FOR USER INPUT FROM ALTERNATIVE TOUCHPADS OF A HANDHELD COMPUTERIZED DEVICE," filed on May 5, 2013, and is a continuation-in-part of U.S. patent application Ser. No. 14/260,195, entitled "METHOD FOR USER INPUT FROM ALTERNATIVE TOUCHPADS OF A HANDHELD COMPUTERIZED SYSTEM," filed on Apr. 23, 2014, which claims priority, under 35 USC §119(e), from U.S. Provisional Patent Application No. 61/815,058, entitled "METHOD FOR USER INPUT FROM ALTERNATIVE TOUCHPADS OF A HANDHELD COMPUTERIZED DEVICE," filed on Apr. 23, 2013, and is a continuation-in-part of U.S. patent application Ser. No. 13/770,791, entitled "METHOD FOR USER INPUT FROM ALTERNATIVE TOUCHPADS OF A HANDHELD COMPUTERIZED DEVICE," filed on Feb. 19, 2013, which is a continuation-in-part of U.S. Pat. No. 8,384,683 B2, entitled "METHOD FOR USER INPUT FROM THE BACK PANEL OF A HANDHELD COMPUTERIZED DEVICE", filed on May 4, 2010, which claims priority, under 35 USC §119(e), from U.S. Provisional Patent Application No. 61/327,102, entitled "METHOD, GRAPHICAL USER INTERFACE, AND APPARATUS FOR USER INPUT FROM THE BACK PANEL OF A HANDHELD ELECTRONIC DEVICE," filed on Apr. 23, 2010, the contents of all of which are incorporated herein by reference in their entirety. U.S. patent application Ser. No. 13/770,791 referenced above is also a continuation-in-part of U.S. patent application Ser. No. 13/223,836, entitled "DETACHABLE BACK MOUNTED TOUCHPAD FOR A HANDHELD COMPUTERIZED DEVICE", filed on Sep. 1, 2011, which is a continuation-in-part of U.S. Pat. No. 8,384,683 B2, entitled "METHOD FOR USER INPUT FROM THE BACK PANEL OF A HANDHELD COMPUTERIZED DEVICE", filed May 4, 2010, the contents of all of which are incorporated herein by reference in their entirety.

BACKGROUND

The present disclosure generally relates to a computerized system including a touchpad for finger actuated control inputs. More particularly, the present disclosure relates to a method that enables the user to use multi-touch gesture controls characterized in three-dimensions.

Handheld computerized devices (i.e. devices including microprocessors and sophisticated displays) such as cell phones, personal digital assistants (PDA), game devices, tabletPCs (such as iPad), wearable computerized devices, and the like, are playing a more and more important role in everyday life, and are becoming more and more indispensible. With the advance of technology, and improvements in the handheld computerized devices' processing power, both function, and memory space is increasing at an amazing pace. Meanwhile the size of the handheld computerized devices continues to get smaller and smaller making the touchpad and display on the device smaller and more challenging to use.

To meet the challenge of a smaller device display and touchpad, the designers of handheld computerized devices typically use two approaches. One approach is to make the keyboard keys smaller and smaller, miniaturizing the keys. Additionally the keyboard keys may be given multiple functions—i.e. overloaded, and more complex function keyboard keys may be introduced as well.

The other approach is to use touch screen keyboards, or so called "soft keys", on the front panel. Here a user may use a stylus pen or finger to select the soft keys through a graphical user interface. However due to the optical illusions introduced by the display screen, and the fact that the user's fingers often are on top of the various display screen soft keys (hence blocking the keys from direct viewing), the soft keys should not be too small. Another problem is that when the soft keys are too small, often a single finger press will activate multiple keys. As a result, the designer may have to divide the keys into different groups and hierarchies, and just display a small number of keys at a time on the screen.

Both current approaches have some drawbacks: the user input area may occupy a significant portion of the front panel, and the user input process, although requiring a large amount of user attention to operate, still is very error prone.

Often a user may use one hand to hold the handheld computerized device, and use the other hand to input data, thus occupying both hands. A user will often have to go through a long sequence of key strokes, and switch back and forth among different user interface screens, in order to complete a fairly simple input. As a result, there is a significant learning curve for a user to learn the overloaded keys, function keys, key grouping, and key hierarchies in order to operate the handheld computerized devices efficiently.

Previous designs including sensors on the back of the device and representations of the user's fingers on the front of the device, however, this work failed to adequately describe a procedure by which the indicia of the user's fingers or hands are displayed on the display panel.

Systems have been described in which image sensors would obtain an image of the user's fingers while operating the device, and use this image data to better determine which real or virtual keys the user's fingers were striking. Such methods rely, however, on image sensors that are positioned in such a way as to be capable of viewing the tips of the user's fingers. This type of image sensor placement is often difficult to implement on many types of handheld user computerized devices. Another drawback of the previous image sensor approach is that it is difficult to implement in low light situations. This approach may also be difficult to implement in situations where there is limited smooth and flat desk or table space.

Conventional touchpads and touchpad controls have used contact areas on the touchpads to form the basis of input and input analysis. These conventional touchpads have entirely

ignored the information available through analysis of epidermal prints which exist within the contact areas.

BRIEF SUMMARY

According to one embodiment of the present invention, a method for controlling an input from a user to a computerized system including a touchpad is presented. The method includes obtaining data from the touchpad. The data is associated with the location and movement of a finger and/or a hand of the user and not associated with an image of the finger from an image sensor, when the user operates the computerized system using the touchpad. The method further includes communicating the data from the touchpad to the computerized system and analyzing the data in accordance with a model of a human hand. The method further includes determining, using the model, that at least one finger of the user is positioned above but not touching a first control region of the touchpad.

According to one embodiment, the method further includes assigning the data to at least one of a multitude of fingers of the model, and computing a first graphical representation of the at least one finger of the user in accordance with the model. The method further includes displaying the first graphical representation on a display screen of the computerized system.

According to one embodiment, analyzing the data and assigning the data includes determining if the data may be assigned to one or more finger or hand portions on the model according to a rotation operation upon at least a portion of the data. According to one embodiment, the method further includes generating an audible signal when the at least one finger of the user is positioned above but not touching the first control region.

According to one embodiment, the method further includes determining, using the model, that a multitude of fingers of the user are positioned above but not touching a first control region of the touchpad. According to one embodiment, the method further includes determining that the multitude of fingers are subsequently touching the first control region in accordance with the data and the model after the multitude of fingers of the user are positioned above but not touching the first control region. According to one embodiment, the method further includes determining a motion of a first finger in relation to a motion of a second finger different than the first finger, and assigning a command to control the computerized system in accordance with the determined motion.

According to one embodiment, the method further includes determining that the at least one finger is initially touching the first control region in accordance with the data and the model before the at least one finger is positioned above but not touching the first control region. According to one embodiment, the method further includes generating a first graphical representation on a display screen of the computerized system, the first graphical representation being associated with the first control region. The method further includes displaying the first graphical representation with a first appearance when the at least one finger is initially touching the first control region, and displaying the first graphical representation with a second appearance different than the first appearance when the at least one finger is positioned above but not touching the first control region.

According to one embodiment, the second appearance includes a difference from the first appearance, the difference being associated with at least one of a size, a color, a position, a shape, or a display type. According to one embodiment, the displaying the first graphical representation with a second appearance occurs for a first period of time less than or equal to 10 seconds.

According to one embodiment, the method further includes determining that the at least one finger is subsequently touching the first control region in accordance with the data and the model after the at least one finger is positioned above but not touching the first control region. According to one embodiment, the method further includes generating a first graphical representation on a display screen of the computerized system. The first graphical representation is associated with the first control region. The method further includes displaying the first graphical representation with a first appearance when the at least one finger is positioned above but not touching the first control region, and displaying the first graphical representation with a second appearance different than the first appearance when the at least one finger is subsequently touching the first control region.

According to one embodiment, the second appearance includes a difference from the first appearance, the difference associated with at least one of a size, a color, a position, a shape, or a display type. According to one embodiment, the touchpad is located in a location that is different from the location of the display screen. According to one embodiment, the touchpad is located in a location that is substantially the same as the location of the display screen. According to one embodiment, the method further includes storing a record that the at least one finger is subsequently touching the first control region. According to one embodiment, the method further includes storing a record that the at least one finger is subsequently touching the first control region within a first period of time less than or equal to 10 seconds after the at least one finger is positioned above but not touching the first control region.

According to one embodiment, the method further includes determining that the at least one finger is moving or sliding on the first control region after subsequently touching the first control region within a first period of time less than or equal to 10 seconds after the at least one finger is positioned above but not touching the first control region, and storing a record of the moving or sliding of the at least one finger. According to one embodiment, the method further includes determining that the at least one finger is moving or sliding on the first control region after subsequently touching the first control region, and storing a record of the moving or sliding of the at least one finger. According to one embodiment, the method further includes generating a first graphical representation on a display screen of the computerized system, the first graphical representation being associated with the first control region. The first control region includes a length substantially greater than the longitudinal length of a surface region of the at least one finger when contacting the touchpad.

According to one embodiment, the computerized system is a handheld computerized device, a computerized system in a vehicle, or a wearable computerized device. According to one embodiment, the computerized system is a handheld computerized device, and the touchpad is located on a side of the handheld computerized device that is different from the side of the handheld computerized device that displays the display screen. The display screen is non-transparent. According to one embodiment, the method further includes designating a portion of the touchpad temporarily as a non-control region for holding the handheld computerized device without controlling an input when the user touches the non-control region.

According to one embodiment, the method further includes storing the data over a multitude of time intervals to form a history of recent finger positions. The method further includes determining an approximate fingertip location or fingertip identity of a fingertip that is not touching the touchpad at a present time interval according to the history and data from the touchpad obtained at the present time interval. According to one embodiment, the method further includes distinguishing a distance between the at least one finger of the user and a surface of the touchpad, and detecting the location and movement of a finger and/or hand of the user to include in the data when the distance is greater than zero.

A better understanding of the nature and advantages of the embodiments of the present invention may be gained with reference to the following detailed description and the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. **1** depicts a simplified exemplary front panel view of the handheld computerized device displaying the position and motion of the user's fingers holding the back panel, in accordance with one embodiment of the present invention.

FIG. **2** depicts a simplified exemplary back panel view of the handheld computerized device depicted in FIG. **1**, in accordance with one embodiment of the present invention.

FIG. **3** depicts a simplified exemplary front panel view of the handheld computerized device depicted in FIG. **1** displaying a multitude of groups of keys, in accordance with one embodiment of the present invention.

FIG. **4** depicts a simplified exemplary front panel view of the handheld computerized device depicted in FIG. **1** displaying the position and motion of the fingers holding the back panel and the multitude of groups of keys depicted in FIG. **3** in the same time, in accordance with one embodiment of the present invention.

FIG. **5** depicts a simplified exemplary front panel view of a smaller handheld computerized device displaying the position and motion of at least one finger in contact with the touchpad of the back panel, in accordance with one embodiment of the present invention.

FIG. **6** depicts a simplified exemplary front panel view of the smaller handheld computerized device depicted in FIG. **5** displaying the position and motion of at least one user's finger in contact with the touchpad of the back panel at the touchpad touch points and a multitude of groups of virtual keyboard keys similarly depicted in FIG. **3** in the same time, in accordance with one embodiment of the present invention.

FIG. **7** depicts a simplified exemplary front panel view of the handheld computerized device displaying another embodiment of the layout of virtual keys as the standard virtual keyboard, in accordance with one embodiment of the present invention.

FIG. **8** depicts a simplified exemplary block diagram of a computerized system capable of executing various embodiments of the invention, in accordance with one embodiment of the present invention.

FIG. **9** depicts a simplified exemplary flowchart how biomechanical models of hand and finger movement may be calibrated and adapted to help turn the raw touchpad data into an accurate model of the user's hand and finger positions, in accordance with one embodiment of the present invention.

FIG. **10** depicts a simplified exemplary flowchart how predictive typing methods may be used to improve the accuracy of the appearance of the virtual hand and fingers while typing, in accordance with one embodiment of the present invention.

FIG. **11** depicts a simplified exemplary flowchart how dynamic changes in touchpad sensitivity may, for finger proximity touchpads, assist in highlighting the virtual keys about to be struck by a user while typing on the virtual keyboard, in accordance with one embodiment of the present invention.

FIG. **12** depicts a simplified exemplary flowchart for generating images of the virtual hand and fingers on the device's graphics display screen, in accordance with one embodiment of the present invention.

FIG. **13** depicts a simplified exemplary biomechanical and/or anatomical model of the human hand, showing the internal skeletal structure with a skin overlay, in accordance with one embodiment of the present invention.

FIG. **14** depicts how the simplified exemplary user's hand or hands may be photographed by the device's camera or other camera, and this image information may be used to refine the default parameters of the biomechanical and/or anatomical model of the user's hand, in accordance with one embodiment of the present invention.

FIG. **15** depicts how an exemplary device camera may be used to obtain a partial image of the user's hand while using the device's touchpad, and this information also used to update and refine the biomechanical and/or anatomical model of the user's hand, in accordance with one embodiment of the present invention.

FIGS. **16**A-**16**B depict how a simplified exemplary palm angle rotation transformation may help the system relate raw touchpad data to a standard biomechanical and/or anatomical model of the human hand, in accordance with one embodiment of the present invention.

FIG. **17** depicts more exemplary details of the relationship between the finger roots and the hand's overall palm angle, in accordance with one embodiment of the present invention.

FIG. **18** depicts more exemplary details of the relationship between the hand's palm direction or palm angle and the tips of the user's fingers, in accordance with one embodiment of the present invention.

FIG. **19** depicts how simplified exemplary biomechanical and/or anatomical model data pertaining to the width of the fingers may be used to help interpret raw touchpad data, in accordance with one embodiment of the present invention.

FIG. **20** depicts how in a more accurate exemplary model, the location of the various finger roots will be displaced to some extent from the palm line (which forms the palm angle) by various amounts δri, in accordance with one embodiment of the present invention.

FIG. **21** depicts how the simplified exemplary system may attempt to correlate detected fingertip data from some fingers with finger root data from other fingers, determine that some fingertip data is missing, and thus deduce that these fingers are elevated above the touchpad, in accordance with one embodiment of the present invention.

FIG. **22** depicts how the simplified exemplary system may further assign raw touchpad data to two different hands of the same user, based on the assumption that the range of possible hand angles for the same user is limited by the user's anatomy, in accordance with one embodiment of the present invention.

FIG. **23** depicts a first simplified exemplary example of angle based finger matching algorithms, in accordance with one embodiment of the present invention.

FIG. **24** depicts a second simplified exemplary example of angle based finger matching algorithms, in accordance with one embodiment of the present invention.

FIG. **25** depicts a simplified exemplary flowchart how biomechanical models of hand and finger movement may be used to display a virtual image of at least a portion of a hand of a user on a display screen of the computerized system of FIG. **8**, in accordance with one embodiment of the present invention.

FIG. **26** depicts a simplified exemplary flowchart of a "lift and tap" technique of key entry for controlling an input from a user to the computerized system, in accordance with one embodiment of the present invention.

FIGS. **27**A-**27**F depict a series of simplified exemplary display screen shots of the "lift and tap" technique of key entry depicted in FIG. **26** being used to type the first two letters of a "Hello World" message on the computerized system, in accordance with embodiments of the present invention.

FIG. **28** depicts a simplified exemplary flowchart of a "lift and drag" technique of key entry for controlling an input from a user to the computerized system, in accordance with one embodiment of the present invention.

FIG. **29** is an illustration of a hand **2900** contacting a touch panel at multiple contact areas.

FIG. **30** is an illustration of a contact area.

FIG. **31** is an illustration of a pattern of contact areas generated by fingertips contacting a touch panel.

FIGS. **32** and **33** illustrate multiple finger configurations which could generate the contact pattern of FIG. **31**.

FIGS. **34**-**39** illustrate correlation between finger contact portions and finger rotation.

FIG. **40** is an illustration of contours of a hand print.

DETAILED DESCRIPTION

The embodiments of the present invention relate to a handheld computerized device including a bit mapped display screen on the front panel, and a touchpad installed on the back panel, side panel, or other area other than that of the display screen. More particularly, the embodiments of the present invention relate to a method and graphical user interface that enable the user to see the user's finger position and motion from behind the device superimposed upon a virtual keyboard layout on the front panel.

It is therefore desirable to have a more efficient and user-friendly way to do user input for handheld computerized devices. The embodiments of the present invention present an effective solution for these above problems. The embodiments of the present invention free the original keyboard space on the front panel for applications by utilizing the previously mostly unused back panel space for user input. The embodiments of the present invention are able handle both keyboard input and mouse input. The embodiments of the present invention present a stunning graphic user interface on the front panel screen where a user may see the real-time position and motion of his/her fingers holding the back panel, on top of the display of keyboard layout (virtual keyboard). The embodiments of the present invention are more precise than current touch screen keyboards by removing the display layer that presently exists between the fingers and touch pad. The embodiments of the present invention also move the user's fingers away from the front panel, so that the user's fingers will not block the view of the soft key or area that the finger is presently operating on. For smaller handheld devices, such as cell phone, iPhone™ or iPad™, the hand that holds the device may now also do input, hence freeing the other hand for other activities.

Thus an object of the embodiments of the present invention are to provide a method for a more efficient and user-friendly user input for a handheld computerized device.

Another object of the embodiments of the present invention are to free up the space currently occupied by the keyboard on the front panel of small electronic devices, and utilize the mostly unused space on the back panel of the handheld devices for user input purposes.

Another object of the embodiments of the present invention are to present a visually compelling user-interface design that enables the real time position and motion of the fingers that hold the device, which normally would be hidden from view by the device itself, to be displayed on the front panel as "virtual fingers" together with an optional display of a virtual keyboard layout. The user's finger positions and keyboard layout may be displayed either as background image, or as a transparent layer on top of some of all of the applications currently running on the handheld device. These semi-transparent representations of the user's finger positions and virtual keyboard allow the user to easily enter data while, at the same time, continuing to allow the user unimpeded access to the various applications running on the handheld device. Thus, for example, applications originally written for a computer device that had a physical keyboard may be easily run, without code modification, on a tablet computer device that lacks a physical keyboard. Thus these virtual semi-transparent keyboards and methods that also give information of finger motion of the user may be highly useful.

Another object of the embodiments of the present invention are to enable the hand that is holding the device to also do user input operations, hence freeing the other hand for other inputs or other purposes.

According to one embodiment, a device and method include a display screen on the front panel, which may be a bit-mapped display screen, a touchpad embedded on the back panel capable of sensing the user's finger positions and motion, and a graphical user interface. This graphical user interface will normally include both software and optional graphics acceleration hardware to enable complex graphics to be rapidly displayed on the display screen. The device also has an optional virtual keyboard processor that displays the keyboard layout, as well as computes and displays the user's finger positions on a real-time basis. The user's finger position and motion on the touchpad of the back panel may thus be computed and displayed on the front display screen as a layer, which may be a semi-transparent layer, on top of all of the other applications. The virtual keyboard processor may also interpret the finger motions, i.e. strokes, and invoke corresponding operations based on the known location of the finger position on the keyboard.

Unlike previous approaches, the user's fingers do not need to be constrained to fit onto particular regions of the touchpad, but rather may be disposed in any arbitrary location. Unlike some previous approaches, although embodiments of the invention may be aided to some extent by real-time video that may provide video information pertaining to at least some portion of the user's hand, visualization of the user's fingers, in particular the tips of the user's fingers is not necessary. This makes it feasible to use handheld device video cameras designed for general photographic purposes to be used to help in visualizing the user's hand, without requiring that much of the user's hand in fact

be photographed. There is no requirement at all that the user's fingertips be photographed while operating the device.

FIG. **1** depicts a simplified exemplary front panel view of a handheld computerized device (**100**) displaying the position and motion of the user's fingers (**108**) holding the back panel, in accordance with one embodiment of the present invention. The user is holding the handheld electronic device **100**, similar to an Apple iPad™ or equivalent pad device. The front panel of the device is occupied by a large graphics display screen (**102**), which may be a bit-mapped graphics display screen. In some embodiments, the whole front panel screen or front panel may be occupied by this graphics display screen (**102**). The user is holding the handheld computerized device (**100**) using his or her hands (**104**), where a portion of the user's thumb (**106**) is in front of the device over a portion of the front panel, and the user's fingers (**108**) are behind the device. Although device (**100**) is not transparent, nonetheless the graphics display screen (**102**) is shown representing a graphical representation of the user's fingers (**108**) as well as regions where the user's fingers are apparently touching an obscured from view or "invisible" surface at touchpad touch points (**110**) at the back panel of the device. Each of the touchpad touch points (**110**) may correspond to a real time finger print image of the tip of the user's finger.

FIG. **2** depicts a simplified exemplary back panel view of the handheld computerized device (**100**) depicted in FIG. **1**, in accordance with one embodiment of the present invention. In contrast to the front panel of device (**100**), previously depicted in FIG. **1**, which included a large graphics display screen, the back panel of the handheld computerized device as depicted in FIG. **2** does not include a large graphics display screen, but instead includes a large touchpad (**200**). As may be seen, the user's fingers (**208**) may now be seen positioned above the touchpad with the tips of the user's fingers (**210**) touching the touchpad.

Note that in some embodiments, this back touchpad may be provided as a retrofit or add-on to a handheld computerized device that otherwise lacks such a back touchpad. Such methods and systems, such as "clip on" back touchpads, are described at more length in patent application Ser. No. 13/223,836, the contents of which are incorporated herein by reference in its entirety.

FIG. **3** depicts a simplified exemplary front panel view of the handheld computerized device depicted in FIG. **1** displaying a multitude of groups of keys (**300**, **302**, **304**), in accordance with one embodiment of the present invention. FIG. **3** depicts one possible optional multitude of groups of keys, i.e. a "virtual keyboard," being displayed on graphics display screen (**102**) of device (**100**). In this example, the "virtual keyboard" includes a symbol keypad (**300**), a numeric keypad (**302**), and a QUERTY keypad (**304**). Note that in many embodiments, the keys may be drawn in outline or semi-transparent form so as not to obscure any other graphical applications running on graphics display screen (**102**).

The scheme depicted in FIG. **3** allows the user to optionally use a touchpad keypad on the back of the device to input keystrokes and mouse actions, and these inputs will be reflected on the display screen on the front of the handheld computerized device as "virtual fingers" or equivalent. As previously discussed, this virtual keyboard layout displayed on graphics display screen (**102**) at the front panel may be a standard or modified QUERTY keyboard or keypad, a numeric keyboard or keypad, i.e. number entry keyboard, or alternatively some less standard keyboard or keypad such as a musical keyboard, a Qwerty, Azerty, Dvorak, Colemak, Neo, Turkish, Arabic, Armenian, Greek, Hebrew, Russian, Moldovan, Ukranian, Bulgarian, Devanagari, Thai, Khmer, Tibetan, Chinese, Hangul (Korean), Japanese, or other type of keyboard. Often this keypad will be a semi-transparent keypad in order to allow the user to continue to view various application programs that are running on display screen (**102**) below the virtual keyboard.

FIG. **4** depicts a simplified exemplary front panel view of the handheld computerized device (**100**) depicted in FIG. **1** displaying the position and motion of the user's fingers (**108**) holding the back panel and the multitude of groups of keys (**300**, **302**, **304**) depicted in FIG. **3** in the same time, in accordance with one embodiment of the present invention. FIG. **4** depicts an example of how a user, typing on a touchpad mounted on the back of the electronic device, may see a graphical representation of his or her fingers (**108**) displayed on graphics screen (**102**) of device (**100**), as well as a display of virtual keyboard layout (**300**, **302**, **304**). The user's ability to enter input data to the handheld computerized device (**100**) is thus enhanced because the user may visually judge the distances between his or her fingers (**108**) and the keypad keys of interest (**300**, **302**, **304**) and move his or her fingers appropriately so as to hit the desired key. The user may also click on hyperlinks, such as link1, link2, and the like, or other clickable objects or command icons.

Because the user's operating fingers are moved away from the display screen, the fingers will not block the view of the display screen's soft keys, soft buttons, links or other areas. These areas on the display screen may now be seen more precisely, which in turn allows for more precise operation of the device.

The virtual display of the user's fingers may be a valuable feature for some of the newer tablet computers, such as the Microsoft Surface™ series, Windows 8, and the like, which may alternate operating modes between a first tablet operating mode designed for traditional touch input, and a second desktop operating mode, derived from legacy desktop operating systems, that is optimized for more precise mouse input. By enabling such tighter control, it becomes more feasible for a user to operate such "Surface" like devices in legacy desktop mode without the need to use a mouse or other hand operated pointing instrument.

Because a front keyboard, i.e. mechanically actuated keys, is no longer necessary, the embodiments of the present invention free up the space on the device that might otherwise have been used for original mechanical keyboard space on the front panel, and create room for additional larger displays and applications. The embodiments of the present invention make use of the presently mostly unused back panel space, thus enabling the front display to show substantially larger virtual keys, or virtual keys including more space between them that are easier for the user to use.

The embodiments of the present invention may create compelling visual effects, as well as useful visual effects, because the user may see his or her fingers (**108**), which are holding the back panel and thus normally blocked from view, being virtually displayed on the front panel along with a virtual, i.e. computer generated, keyboard layout display (**300**, **302**, **304**). Because both the user's finger position, finger touch area, each depicted as a circle surrounding a cross, finger motion and the virtual keyboard are visible from the front panel, the user finger inputs on the touch panel located on the back panel of the device are both intuitive and easy to use. There will be no learning curve, and no need for special training. The user input methods of the embodiments of the present invention are more precise than traditional

touch screen keyboards because these methods remove the obscuring layer between the finger and touchpad, and the operating fingers will not block the view of the display screen. For small handheld devices such as cell phones and iPhones, the current embodiments of the present invention enable the hand that holds the device to perform text input and other commands, hence freeing the other hand for other activities.

Note that although often a virtual keyboard will be presented, alternative data entry points of interest, such as hyperlinks on an internet browser, and the like, may also be used according to these methods as well.

In one embodiment, the layout of a multitude of groups of virtual keyboard keys (**300**, **302**, **304**), including numbers, letters, and symbols may be displayed on an area separated from concurrently running other software applications that are being displayed simultaneously on the screen of the front panel (much like the traditional separately displayed area often used for soft keys near the bottom of the display screen). The virtual keyboard keys (**300**, **302**, **304**) may be advantageously displayed in different size or in locations that are not the same locations that are determined by the other software applications and/or programs because the virtual keyboard keys (**300**, **302**, **304**) may be displayed translucently so as to display both the virtual keyboard keys (**300**, **302**, **304**) and the underlying concurrently running application or program display content.

Devices and systems utilizing the virtual fingers and optional virtual keyboard embodiments of the present invention advantageously need not have mechanically actuated and/or permanently dedicated physical QUERTY keypads or QUERTY keyboards, or any other type of mechanically actuated and/or permanently dedicated physical keypad or keyboard such as one dedicated to number entry. Eliminating mechanically actuated and/or permanently dedicated keypads or keyboards improves device ergonomics, allow for larger graphics display screens, and also reduces device costs.

FIG. **5** depicts a simplified exemplary front panel view of a smaller handheld computerized device (**500**) displaying the position and motion of at least one finger in contact with the touchpad of the back panel, in accordance with one embodiment of the present invention. Smaller handheld computerized device (**500**) may include a cellular phone sized device (e.g. an Apple iPhone™ sized device) including a smaller graphics display screen (**502**) virtually displaying the position and motion of a multitude of fingers (**108**) in contact with the touchpad touch points (**110**) at the back panel of smaller handheld computerized device (**500**).

FIG. **6** depicts a simplified exemplary front panel view of the smaller handheld computerized device (**500**) depicted in FIG. **5** displaying the position and motion of at least one user's finger (**108**) in contact with the touchpad of the back panel at touchpad touch points (**110**), and a multitude of groups of virtual keyboard keys (**300**, **302**, **304**) similarly depicted in FIG. **3** in the same time, in accordance with one embodiment of the present invention. FIG. **6** may include similar features as FIG. **4** with the exception of using smaller handheld computerized device (**500**) being held in just one user hand (**104**), the other hand of the user being free to do other tasks.

FIG. **7** depicts a simplified exemplary front panel view of handheld computerized device (**100**) depicted in FIG. **1** displaying another embodiment of the layout of virtual keys (**700**) as the standard virtual keyboard, in accordance with one embodiment of the present invention. FIG. **7** depicts another embodiment of the layout of virtual keys (**700**) may include a modified QUERTY keyboard or keypad that includes splitting the keyboard in half and displaying each half of the keyboard at an angle adapted for better ergonomic typing than the keyboards depicted previously.

In one embodiment, a computer-implemented method includes a handheld computerized device, including a screen on the front of the device capable of displaying a graphical user interface, and a touch sensitive back panel (or side panel or other area other than the display screen) user interface, such as a two dimensional touch sensor. For simplicity, this touch sensitive panel, which need not necessarily be flat, and need not necessarily be mounted on the back side of the device, hereinafter also referred to as a "touchpad," "touch sensor," or "touch sensitive back panel", but this use is not intended to be limiting.

The touch sensor will determine the motion of the fingers in real time, and the computerized system's or device's software and processor(s) will use the touch sensor data to compute the real time position and motion of the user's fingers that are touching the touch sensor on the back panel. These "virtual fingers" will then be displayed on the device's graphical user interface on top of a static background where optionally a multitude of groups of keys, including numbers, letters, and symbols (e.g. a virtual keyboard) or hyperlinks may be displayed. By watching the motion of the user's virtual fingers on the virtual keyboard, the user may easily operate the device, and optionally determine precisely where to strike a finger in order to hit an intended virtual key.

In one embodiment, the back panel user interface (UI) may be outlined in a distinctive yet non-obstructive color and displayed as a transparent layer over the current applications; hence all the details of current application and back panel UI are shown to the user at the same time.

In one embodiment, the real time position and motion of the fingers holding the back panel may be displayed on the screen of the front panel.

In one embodiment, the layout of a multitude of groups of keys, including numbers, letters, and symbols may be displayed on the screen of front panel as background of real time position and motion of the fingers holding the back panel.

In one embodiment, the real time position and motion of the fingers holding the back panel may be displayed on the static background of a multitude of groups of keys, including numbers, letters, and symbols, enabling the user to precisely strike a finger on an intended key.

In one embodiment, the display of the virtual hand may be creative and artistic. For example, the display may instead show a skeleton, an animal claw, a furry hand, a tattooed hand, and the like to achieve more compelling or amusing effects.

In one embodiment, a computer-implemented method, including a handheld computerized device includes a touchpad installed on the back panel. The touchpad is able to sense the touch point positions, movement, and stroke motion data of a multitude of fingers. The data information of the finger motion of one or a multitude of fingers, including the motion type (e.g., touch, movement, and stroke patterns, and the like), and motion position, is passed to a virtual keyboard processor (such as a computer processor). The virtual keyboard processor may analyze the finger motion, compare the finger positions with the registered position of the keys (virtual keys) as well as the hyperlinks and other touch buttons of the application program (generically the "user entry area"), and then will decide which item in the user entry area was stroked or actuated. The virtual

keyboard processor may then invoke the corresponding operation. The virtual keyboard processor may also update the real time image of the fingers, or finger pads (touch points), or indeed the user hand(s) on the front screen after each finger motion.

In one embodiment, the touchpad may be installed on the back panel of the handheld computerized device, and may be able to sense the touch, movement, and stroke motion of a multitude of user fingers.

In one embodiment, the information pertaining to the finger motion of a multitude of user fingers, including the motion type (e.g., touch, movement, and stroke action, and the like), as well as motion position, may be passed to a virtual keyboard processor.

In one embodiment, the virtual keyboard processor may analyze the finger motion, compare finger position with the registered position of the keys, determine which key was stroked, and invoke the corresponding operation,

In one embodiment, virtual keyboard processor may update the real time position and motion of the fingers holding the back panel.

One embodiment of the present invention includes a graphical user interface (GUI) for a handheld computerized device. The interface may include a display of a multitude of groups of keys, including numbers, letters, and symbols. The keys may be displayed on a graphical user interface on the front panel display screen, and indeed this display area may occupy the whole screen. Thereby, the content of the graphic user interface is not blocked by applications, and is shown together with the applications.

One embodiment of the present invention includes a graphical user interface for a handheld computerized device. This interface includes a display of the real time position and motion of the fingers holding the back panel. Here the display is on the front panel screen, and in fact may occupy the whole screen. Due to the advantages of this approach, the content of the user's finger position and motion is not blocked by applications, or by the display of groups of keys, including numbers, letters, and symbols.

One embodiment of the present invention includes a method of assisting user data entry into a handheld computerized device. This handheld computerized device includes at least one touchpad (in one embodiment being located on a side of the handheld computerized device that is behind the side of the device that holds the graphics display screen), at least one graphics display screen, at least one processor, memory, and software. Often, however, the handheld computerized device will lack a mechanically actuated and/or permanently dedicated physical QUERTY keypad or keyboard, and may also lack a mechanically actuated and/or permanently dedicated physical numeric keypad or keyboard as well. The method will usually include displaying at least one data entry location on the at least one graphics display screen of the device. Often this at least one data entry location will be a graphical display of a keyboard or keypad that may be included of a multitude of data entry locations. Here, the system will use the touchpad to obtain data on the location and movement of the user's fingers and/or hand. The system may analyze the data on the location and movement of the user's fingers and/or hand according to a biomechanical and/or anatomical model of a human hand, and will assign data on the location and movement of the user's fingers and/or hand to specific fingers on this biomechanical and/or anatomical model of a human hand (usually the user's hand). The system may then use this biomechanical and/or anatomical model of the human hand to compute a graphical representation of at least the user's fingers, and frequently both the user fingers and the user hand(s). The system will then display the graphical representation of at least the user's fingers (and again frequently both the user's finger and hand), on the device's graphics display screen. Thus the distance between the graphical representation of the user's virtual fingers on the graphics display screen, and the virtual data entry location (such as the virtual keyboard) will give information that will help the user properly position his or her real fingers and/or hand on the touchpad, which in turn will facilitate data entry.

FIG. **8** depicts a simplified exemplary block diagram of a computerized system **800**S capable of executing various embodiments of the invention, in accordance with one embodiment of the present invention. Computerized system **800**S includes software and hardware that may be used to implement one embodiment of the invention such as a front panel screen (**804**), a back panel touch pad **800**, a virtual keyboard processor (**802**), an application process (**806**), and a device memory (**808**). Finger position and motion data are first collected from back panel touch pad (**800**), and then passed to virtual keyboard processor (**802**). The virtual keyboard processor (which will often be implemented by a combination of software and hardware such as a microprocessor, graphics processor, touchpad controller, and memory) displays the virtual finger position and motion together with the keyboard layout on front panel screen (**804**). The virtual keyboard processor also analyzes the finger position and motion information data, compares the data with the registered position of the keys (or hyperlinks) and invokes proper operation in application process (**806**). The keyboard position information may be programmed in a virtual keyboard process, or alternatively may be saved in system memory (**808**). The key-press or hyper-link information that the user intends to relay to the applications may be passed to the virtual keyboard controller either through memory, or alternatively through inter-process communications.

Other Touchpad and Screen Locations.

In one embodiment, the display screen may be located at some distance from the touchpad. Indeed, the display screen and the touch pad may not even be physically connected at all. Rather the touchpad may transmit data pertaining to the user's hand position to a processor, which in turn may then generate the virtual image of the user's hand and display the virtual hand on the display screen, and neither touchpad, processor, or display screen need to be physically connected (although they may be). For example, data pertaining to the user's hand and finger position relative to the touchpad may be transmitted by a wired, wireless, or optical (e.g. infrared) method to the processor. The processor in turn may transmit the virtual image of the user's fingers and hand to the display screen by a wired, wireless, or optical (e.g. infrared) technique. As a result, the user's real hand will be moving close to a touchpad at a different place other than the current display screen. The display screen may thus be in nearly any location, such as on a regular monitor, TV screen, projector screen, or on a virtual heads-up eyeglass display worn by the user (e.g. a device similar to Google Glass).

Touch Pads Including Non-Flat Surfaces.

Although touch pads are often flat and roughly rectangular devices, there is no constraint that the touch pads using embodiments of the present invention be either flat or rectangular. Indeed in some embodiments, there is advantage to employing touch pads that include variably shaped and curved surfaces. Such curved and/or variably shaped touch pads could be then placed on various non-traditional locations, such as on the surface of a ball or cylinder, on the

surface of various common devices such as glasses frame stems for virtual heads-up displays such as windshields, eyeglasses, and the like, other wearable computerized devices such as smart watch bands, steering wheels—either for a vehicle or a game interface, joysticks, and the like, and/or, dashboards, instrument panels, and the like.

Touchpad Technology.

In principle, many different types of touchpad technology may be used for this device, including capacitive sensing, conductance sensing, resistive sensing, surface acoustic wave sensing, surface capacitance sensing, projected capacitance sensing, strain gauges, optical imaging, dispersive signal technology, acoustic pulse recognition, pressure sensing and bidirectional screen sensing. However, in a preferred embodiment, touchpad sensing technology that is capable of sensing multiple finger positions at the same time may be used. Such an ability to sense multiple finger positions or gestures at the same time hereinafter also referred to as "multitouch" or "multi-touch" sensing technology. Touchpads are thus distinguished from previous mechanical keyboards or keypads because touchpads are not mechanically actuated, that is, since the surface of a touchpad is substantially rigid and responds to touch instead of a mechanical deflection, the touchpad gives the user substantially no indication that the immediate surface of the touchpad moves where touched, except perhaps for the entire rigid touchpad moving as a result, even with pressure sensitive touchpad technology. Touchpads are further distinguished from previous mechanical keyboards or keypads because the shape and/or location of input keys or buttons on a touchpad are not fixed because the keys and/or buttons are instead displayed on an electronically controlled screen with the flexibility of software control and not limited by fixed mechanical elements located on the device.

One example of a multi-touch touchpad embodying the present invention may use a touch sensing device commercially available from Cypress Semiconductor Corporation, San Jose, Calif. and commonly known as the Cypress TrueTouch™ family of products. This family of touchpad products works by projective capacitive technology, and is suited for multi-touch applications. The technology functions by detecting the presence or proximity of a finger to capacitive sensors. Because this touchpad system senses finger proximity, rather than finger pressure, it is well suited to multi-touch applications because, depending upon the tuning of the capacitance detection circuit, various degrees of finger pressure, from light to intense, may be analyzed. Although often used on touch screens, the projective capacitive technology method may function with a broad range of substrates.

Virtual Finger and Hand Position Software (Virtual Keyboard Processor)

As others have noted, one problem with attempting to create "virtual fingers" is that at best, usually just certain regions of the hand, such as the fingertips and perhaps the palms, may usually be detected by conventional multi-touch sensors. To overcome this issue bootstrapping from hand-position estimates has been suggested, which overcomes the invisibility of structures that link fingertips to palms. Suitable algorithms could be obtained by using context-dependent segmentation of the various proximity image constructs, and by parameterizing the pixel groups corresponding to each distinguishable surface contact. It was found that by path-tracking links across successive images, those groups which correspond to the same hand part could be determined, and it was possible to reliably detect when individual fingers touched down and lifted from the multi-touch pad surface. It has been proposed that a number of different combinatorial optimization algorithms that used biomechanical constraints and anatomical features to associate each contact's path with a particular fingertip, thumb, or palm of either hand. Such algorithms further operated by assigning contacts to a ring of hand part attractor points, using a squared-distance cost metric, to effectively sort the contact identities with respect to the ring of hand part attractor points.

A skeletal linked model of the human hand based software that creates a biology-based (biomechanical and/or anatomical) model of joint motion and associated set of constraints has been proposed. The skeletal linked model approach also is based on a software model of the skin that may stretch and bulge in order to accommodate this internal skeleton. The software models a natural joint axis for four different types of joints in the human hand, as well as takes into account the relative lengths of the underlying hand bone structure, and also accounts for the space occupied by the hand's muscles and skin.

FIG. **25** depicts a simplified exemplary flowchart how biomechanical models of hand and finger movement may be used to display a virtual image of at least a portion of a hand of a user on a display screen of the computerized system of FIG. **8**, in accordance with one embodiment of the present invention. FIG. **25** depicts the flowchart includes obtaining data from a touchpad, the data being associated with the location and movement of a finger and/or hand of the user and not associated with an image of the finger of the user from an image sensor, when the user operates the computerized system using the touchpad (**2510**). The flowchart further includes communicating the data from the touchpad to the computerized device, the touchpad being located in a location that is different from the location of the display screen (**2520**). The flowchart further includes analyzing the data in accordance with a model of a human hand, and assigning the data to at least one of a multitude of fingers of the model (**2530**), computing a graphical representation of at least one finger of the user in accordance with the model (**2540**), and displaying the graphical representation on the display screen (**2550**).

FIG. **9** depicts a simplified exemplary flowchart how biomechanical models of hand and finger movement may be calibrated and adapted to help turn the raw touchpad data into an accurate model of the user's hand and finger positions, in accordance with one embodiment of the present invention. The system may work with adequate accuracy using standardized models of hand and finger relationships. For example, the system may perform adequately by an initial calibration step where the system invites the user to place his or her hand on the display screen, the system displays various sized hands, and the user is invited to enter in which standardized hand size best fits his or her own hands. The system may then use this data for its various calculations. Even more simply, the system may default to an average hand size for initial use, allowing some degree of functionality to be achieved with no preliminary calibration.

In one embodiment, it will be useful to better calibrate the system by employing one or more active calibration steps. These steps may refine the initial hand model under actual use conditions, and make appropriate adjustment to the various portions of the hand model as will best fit data that has been obtained under actual use conditions. An example of this active calibration process is shown in FIG. **9**. Here the system may invite the user to do an active calibration step, or alternatively the user will voluntarily start an active calibration step, in step (**900**). In one embodiment, the model

includes calibration information in accordance with pressing a portion of the user's hand on the touchpad in a specified manner. To facilitate this calibration step, the system may optionally display one or more targets on the screen, which may be keyboard targets, or alternatively may be specially designed calibration targets specifically designed for the active calibration step. Optional photographic calibration steps are described for FIG. **14**.

In one embodiment referring to FIG. **9**, to reduce complexity, the system may optionally request that the user calibrate one hand at a time, and indeed may request that the user operate the fingers on his or her hand in a manner different from normal typing so as to gather additional data. For example, a user may be requested to first extend a specific finger to a maximum length and press, then to a minimum length and press, then to the extreme left and press, then to the extreme right and press and so on, potentially through all fingers and the thumb on a one at a time basis. It should be apparent that such a data set may then naturally be translated into a reasonably detailed model of that particular user's hand and its capabilities to maintain a number of different configurations. During the calibration process, the system will accumulate touch data by invoking touchpad sensing hardware and calibration software (**902**). The system will also make predictions as to the location of the user's hand and fingers by bootstrapping from various hand position estimates (**904**). Often the system will track the positions of the hand and fingers across successive time intervals to do the predicting, and compute probable finger paths (**906**). The system will often use its internal model of the user's hand biomechanical features and anatomical features to do the computing, and to help associate the various projected paths with the user's fingertips and thumb position, which at least during the active calibration process will be known (**908**). Here a path is understood to be the line or linkage between at least one finger root and the associated fingertip or touchpad touch point for the associated finger. The system will then refine its models of the user's hand biomechanical and/or anatomical features by comparing the predicted results with real data, and determine if its user hand model is working with sufficient accuracy in step (**910**). If it is, then this user hand model will then be adopted and used for subsequent user virtual keyboard data entry purposes (**914**). If the user hand model is not working with sufficient accuracy, then the system will attempt to adjust the hand model by varying one or more hand-model parameters (**912**), and often will then continue the calibration process until acceptable performance is obtained.

Thus the calibration software enables the biomechanical and/or anatomical model of the human hand to be calibrated more accurately, so as to match the biomechanical and/or anatomical characteristics of a particular user's fingers and/or hand.

In one embodiment, the realism of the simulated virtual fingers on the screen may optionally be facilitated by the use of predictive typing models. The predictive typing model approach will be particularly useful when the user is typing text on a virtual keyboard, because the system may scan the previous text that has been entered, and utilize a dictionary and other means, such as the statistical distribution of letters in the particular language, to make educated guesses as to what letter is going to be typed next. This educated guess may then be used to supplement the touchpad data as to last fingertip position and movement to tend to direct the appearance of the simulated finger towards the logical next key. Because this system will occasionally tend to guess wrong, however, the user may find it useful to adjust this predictive typing "hint" to various settings depending upon the user and the situation. Thus a user who is an experienced touch typist and who tends to type both fairly quickly and fairly accurately will tend to find the predictive typing hints useful, because the predictive approach will tend to work well for this type of user. On the other hand, a user who is more of a slow and uncertain "hunt and peck" typist may find the predictive approach to be less useful, and may wish to either reduce the strength of the hint or potentially even turn the predictive typing "hint" off altogether.

FIG. **10** depicts a simplified exemplary flowchart how predictive typing methods may be used to improve the accuracy of the appearance of the virtual hand and fingers while typing, in accordance with one embodiment of the present invention. In a predictive typing system, the software will first access both the biomechanical and/or anatomical model data for the user's hands (**1000**), and the latest fingertip and thumb position data from the touchpad sensors (**1002**). The system will then use this information to display the user's virtual hands and fingers on the device's display screen (**1004**). If a predictive typing mode is on (**1006**), then the system will attempt to deduce (based upon typing speed, as well as the user's consistency in typing speed, and context) what is the most probable letter or letters that the user is likely to type next. The system will also attempt to predict the most probable finger or fingers that the user will use to type this most probable letter (**1008**). For example, if the user is typing quickly and consistently, and the context of the word or sentence indicates that a vowel such as "e" is likely, then the system may use this factor in its analysis of the somewhat noisy finger position data from the touch sensor to increase the probability that the user's left index finger (often used to type "e" on a keyboard, and which in-fact may not be registering on the touch pad because the user has lifted the left index finger to move to strike the "e" key), is moving towards the "e" key. When used properly, such predictive typing algorithms may help increase the illusion that the user is looking through the display and onto his or her hands below the display even though the display/computerized device is not actually transparent. Conversely, if the predictive typing mode is turned "off" (or set to reduced intensity) (**1010**), then the system will not take the probable next letter into account in its display of the user's hand and fingers and instead just displays using the virtual hand(s) model.

In one embodiment, the efficiency of the predictive typing may be further enhanced by incorporating the user's history of finger use for each particular key. For example, one user may have a strong tendency to use the right index finger to type the keys "H" and "J", and as another example the same user may have a tendency to user his or her left pinky finger to type the letter's "A" and "Z". Here the system may observe the individual user's typing patterns over time, either as part of an initial calibration step, or later (and in one embodiment even continually) while monitoring the user's typing patterns, and use the user's individualized finger-to-letter correlation habits as part of the predictive typing algorithm.

Thus the predictive typing software enables the computerized device to compute the graphical representation of at least the user's fingers (and often the user's fingers and hands) with better precision by additionally using keystroke predictions, in addition to the data on the location and movement of the user's fingers and/or hand obtained using the touchpad.

In one embodiment, in order to improve the realism of the virtual fingers, additional "finger hover" algorithms may

also be used. As used in this specification, "finger hover" means highlighting or otherwise graphically altering the appearance of a virtual key on a virtual keyboard whenever the system believes that the user's finger is either hovering above that virtual key, or about to strike that virtual key. For this type of algorithm, use of touchpads that may sense relative finger proximity to the touchpad surface, such as projective capacitive technology touchpads, may be particularly useful.

The sensors and algorithms that detect relative finger-height above a surface may be tuned to various degrees of sensitivity, and indeed this sensitivity level represents an important engineering tradeoff. If the touchpad is tuned to too high a sensitivity, then it will tend to generate spurious (false) signals, and also lack precision as to precisely where on the touchpad a finger is about to land. If the touch pad is tuned to a lower sensitivity, then the touchpad will tend to detect fingertips that are exerting a considerable amount of pressure on the touchpad surface.

Although many prior art touchpads tend to use a continual or fixed level of touchpad sensitivity at all times, in one embodiment for the "finger hover" option described in this specification, use of a dynamic or variable level of touchpad sensitivity may be advantageous. For example, to detect finger hovering above a key, a touchpad might first operate at a normal level of sensitivity until it detects that a fingertip within strategic striking distance of a particular key has left the surface of the touchpad. At this point, in order to detect "finger hover" above the key, the touchpad circuitry might temporarily reset its sensitivity to a higher level, designed to more precisely detect when the user's finger is hovering above the key. If the higher level of touchpad sensitivity detects the fingertip proximity, the key may be highlighted. If the higher level of touchpad sensitivity does not detect the hovering fingertip, then the key will not be highlighted. After a short period of time, about on the order a tenth of a second, the touchpad may then be reset to the normal level of sensitivity to more precisely determine if the finger has then actually touched the touchpad, or not.

FIG. **11** depicts a simplified exemplary flowchart how dynamic changes in touchpad sensitivity may, for finger proximity touchpads, assist in highlighting the virtual keys about to be struck by a user while typing on the virtual keyboard, in accordance with one embodiment of the present invention. In other words, FIG. **11** depicts an example of an algorithm to detect and indicate "finger hover". Here the system displays the virtual keyboard (**1100**), as well as an overlay of the user's virtual fingers on or near this virtual keyboard (**1102**). When the system detects that a finger, suspected of being a finger about to press a key due to the finger's proximity to the key and or predictive typing considerations, leaves the touchpad (most likely because the user has raised the finger above the touchpad in preparation for striking the virtual key), (**1104**) the system will momentarily turn the touchpad finger proximity detector to a higher level of sensitivity (**1106**), and the software will look to see if finger hover over the suspected key or keys may be detected (**1108**). If the system does not detect that a finger is suspect of leaving the touchpad, the system returns to step **1102**. If a finger hover signal may be detected over the suspected key, then this key will be highlighted to help guide the user (**1110**). After a period of time that will not normally exceed about a tenth of a second or if no finger hover is detected, the system will once again lower the sensitivity of the finger proximity detector down to the normal level (**1112**), in order to precisely detect if the finger is about to strike the key (**1114**). If the touchpad, now operating at normal sensitivity, now detects that the virtual key has been struck by the user, the system will appropriately indicate the keystroke on the virtual key board by further graphical changes to the key (**1116**) and optionally may issue an audible key-press or key-click sound as well to give further feedback to the user. Then the system may record the key strike (**1118**). If the appropriate finger press was not detected at (**1114**), then the system repeats the flow at step (**1102**).

More generally, the finger hover algorithm approach allows at least one data entry location (key) to be highlighted on the device's graphics display screen whenever the computerized device determines that at least one finger on the user's hand has left the touchpad, and the position and motion history of the finger is consistent with an ability of that finger to strike a position on the touchpad that is consistent with the location of the data entry location (key) on the graphics display screen.

Graphical Representation of the User's Human Hand(s) and Fingers.

Once the computerized device has obtained data from the touchpad, as well as any additional predictive typing data, hover detection method data, calibration data, and the like, and has updated its internal biomechanical and/or anatomical model of the user's hand or hands (including the fingers) to reflect this new data, then the system may utilize this biomechanical and/or anatomical model of the user's hand or hands to compute a graphical representation of at least the user's fingers, and often the user's hand and figures, suitable for display on the device's graphics display screen.

A life-like graphical representation of the user's hand and fingers is not necessary. Often, a more shadow-gram like or cartoon-like two-dimensional model (or representation) of the user's hand and fingers will be all that will be necessary. Often these two-dimensional representations of the user's hand and fingers need not include much, if any internal detail. Rather, these representations, may for example, look much like a translucent gray or other colored shadow projection of the user's hands and fingers on a surface. Here, the sharpness and the contrast and the detail of the user's hands and fingers may have reduced sharpness, and have enough distinguishing contrast from other areas of the display screen, so as to enable the user to accurately place his or her hands and fingers on the appropriate virtual buttons or virtual keyboard that is being shown in the graphical display. More fanciful or artistically inspired hand representations are also discussed later in this specification.

FIG. **12** depicts a simplified exemplary flowchart for generating images of the virtual hand and fingers on the device's graphics display screen, in accordance with one embodiment of the present invention. Many ways to graphically represent the user's hands and fingers, or at least the user's fingers, are possible. In one embodiment, based upon the biomechanical and/or anatomical model of the human hand(s) (**1200**), and optionally specific data on the location and movement of the user's fingers and hand based on the touchpad data (as well as any additional data from predictive typing software, or hover detection) a three-dimensional virtual model may be constructed in the device's memory that depicts the user's hand(s) and fingers (**1202**).

Based upon the 3D model, a two-dimensional projection of the general outlines of the user's hand and fingers may be made upon a mathematical surface that corresponds to the surface of the touchpad (**1204**). This projection may be in the form of a hand and/or finger outline, or alternatively a virtual hand and finger shadow may be produced. This projection may then be combined with the any other data that is being

sent do a memory buffer or graphics display buffer for the display screen of the device, and then displayed to the user (**1206**).

Thus, in one embodiment, the graphical representation of at least the user's fingers (and often both the user's hand and fingers) on the graphics display screen may be done by using the previous assignment of the data on the location and movement of the user's fingers and/or hand(s) to specific fingers on the biomechanical and/or anatomical model of the human hand(s) to create a three dimensional model of the user's hand(s) and fingers in the computerized device's memory. Next, a two-dimensional projection of this three dimensional model of the user's hand(s) and fingers in memory may be made. Here the two-dimensional projection may be on an imaginary plane that corresponds in both distance and orientation from the model of the user's fingers to the touchpad. Thus if, for example, the real user's finger is ¼" above the touchpad, then the distance between the three dimensional model of the user's finger and the imaginary plane that corresponds in distance and orientation to the touchpad will also be ¼". This two-dimensional projection on the imaginary "touchpad" plane (virtual touchpad) may be used to generate the graphical representation of at least the user's fingers on the graphics display screen, and often the user's fingers and hand(s) as well.

Alternatively, in a less computationally intensive scheme, a two dimensional model of the user's hands and fingers may be manipulated to best fit the previously discussed hand and finger position and motion data, and this two dimensional model then used for the graphical representation.

This two dimensional model may be further user selected according to the user's hand size, and indeed may be calibrated by asking the user to place his or her hand on the touchpad, thus allowing the system to sense the dimensions of the user's hand directly.

FIG. **13** depicts a simplified exemplary biomechanical and/or anatomical model of the human hand, showing the internal skeletal structure with a skin overlay, in accordance with one embodiment of the present invention. This illustration shows the major bones of the hand, with the bones of the index finger and thumb separated in order to allow the joints to be better visualized. The internal skeletal structure of the hand (**1300**) is depicted, along with an outline of the skin on the left side of the hand (**1302**). The bones of the fingers include the distal phalanges (**1304**), the intermediate phalanges (**1306**), the proximal phalanges (**1308**) and the metacarpals (**1310**). The thumb lacks the intermediate phalange.

Here the various finger joints include the distal inter-phalangeal joint (dip) (**1312**), the proximal inter-phalangeal joint (pip) (**1314**), and the metacarpophalangeal joint (mcp) (**1316**). The thumb lacks the distal inter-phalangeal joint (dip), and instead includes the interphlangeal joint (ip) (**1318**) as well as the carpometacarpal (cmc) joint (**1320**). In one embodiment for higher accuracy, it may be useful to replace the default parameter values of at least the lengths of these various bones with actual user hand parameters. In general, the closer the various default parameters of the biomechanical and/or anatomical model of the human are to the actual user hand parameters, the better. In some embodiments, even the range of joint motion may also be experimentally determined, and used to replace one or more joint motion range default parameters.

Finger Identifying Algorithms.

In some embodiments, the biomechanical and/or anatomical model of the human hand used in the embodiments of the present invention for finger identifying algorithms may be based on the following observations. First, the average human hand has four fingers and one thumb. Second, in contrast to the fingers and thumb of the human hand (e.g. FIG. **13** bones (**1308**), (**1306**), (**1304**), which are relatively flexible above the metacarpophalangeal joint (mcp) (**1316**), the palm of the average human hand is relatively inflexible below the metacarpophalangeal joint (mcp) (**1316**). Indeed the positions of the various metacarpophalangeal joints (mcp) (**1316**) tend to be relatively invariant with respect to rotation of the hand. The various metacarpophalangeal joints (mcp) (**1316**) may hereinafter also be referred to as the "finger roots". Finger roots will be represented by the variable "r". Alternatively, finger roots may be referred to as the junction between the finger and the palm.

Third, due to the relatively invariant shape of the palm, the orientation of the user's palm and its angle with respect to other hand structures, such as the relative orientation of the fingers (e.g. middle finger (**1330**) is relatively constant. In particular, the orientation or position of the various "finger roots" (**1316**) may define a palm line direction (**1332**) that will in turn, when the angle of the palm line with respect to the coordinates of the touchpad are known, help to define the location of the various fingers and fingertips.

Fourth, users may generally desire to manipulate symbols using the area underneath the uppermost bone of the finger or thumb (**1304**). Here, the touch pad data will include various touchpad touch points, identified in (x, y) coordinates in later figures, which will often but not always correspond to the area underneath the uppermost bone of the user's finger and thumb (**1304**), hereinafter also referred to as the "finger tips". The touchpad observed location of any given finger or thumb tip will often be referred to as (x_i, y_i), where x and y are the observed touchpad data, and "i" refers to or is associated with the finger that ultimately produced the touch pad touch data.

The raw touch pad data does not include such (x_i, y_i) labels. Instead, the system embodiments may have to make sense of various incoming touch pad data, attempt to make sense of the data using the underlying biomechanical and/or anatomical model of the human hand, and then generate a virtual hand model that is consistent with both the touchpad data and the underlying biomechanical and/or anatomical hand model.

It may be simpler to first consider a model of the human hand as it rotates to various positions and touches the touchpad with various fingers, and determine what sort of mathematical transformations are at work in generating the raw touchpad data. Once the above is determined, the process may be worked in reverse to generate a virtual model of the user's hand.

In one embodiment, it is desired to be able to determine which touch point belongs to which finger when the system detects multiple touch points on the touch pad. Naming convention are described as follows. Thumb=finger **0**=F**0**, index finger=finger **1**=F**1**; middle finger=finger **2**=F**2**; ring finger=finger **3**=F**3**, and little finger (pinky)=finger **4**=F**4**. As previously discussed, the finger number may be represented by the variable "i". Thus, when fingers F**0** to F**4** are present on the touch pad, the problem becomes one of determining the coordinates (x_i, y_i) of the fingertips of F**0** to F**4**, and then mapping the coordinates to the biomechanical and/or anatomical model of the human hand.

Neglecting, for the moment, hand location on the touchpad issues, one problem is that users will usually operate the touchpad with the palm line direction of their hands (**1322**) at an arbitrary angle θ with respect to the coordinate system of the touchpad. Thus, an early step to making sense of the

touchpad data is to determine this angle θ, and to transform the raw touchpad data by a rotation of angle θ and see if the raw touchpad data matches up to a sensible biomechanical and/or anatomical model of the human hand. This transformation problem is depicted in FIG. **16**A through FIG. **19**.

In one embodiment, a simplified exemplary palm angle (θ) rotation transformation may help the system relate raw touchpad data to a standard biomechanical and/or anatomical model of the human hand, in accordance with one embodiment of the present invention. If the user touches both the tips of all fingers and thumb and the base (finger root) of all fingers and thumb onto the touchpad, then the raw touchpad data would include a series of (x_i, y_i) values for the finger tips, and a series of (x_{ri}, y_{ri}) values for the finger roots.

In one embodiment, the system may determine how much the user's hand is rotated relative to the coordinate system of the touch pad (palm angle θ) then the process of mapping the raw data into the biomechanical and/or anatomical model of the human hand may be simplified. Thus, a determination of palm angle θ is useful.

FIG. **18** depicts more exemplary details of the relationship between the hand's palm direction or palm angle and the tips of the user's fingers, in accordance with one embodiment of the present invention. When using touchpads, users will often touch the pad with the fleshy area of the palm underneath their finger roots (**1822**). If the finger root touch area information is detected by the touch pad, the system may detect the direction of the palm line of the hands (**1322**) from the finger root touch area. Alternatively, the system may use a relaxed finger position direction depicted as dashed—dotted line (**1825**) from touchpad touch point (**1810**) on F**1** to touchpad touch point (**1820**) on F**3**, or a relaxed finger position direction from touchpad touch point (**1810**) on F**1** to touchpad touch point (**1830**) on F**4** to approximate the palm line direction (**1322**) and adjustment angle σ between the relaxed finger position direction and the palm line direction (e.g. between line (**1825**) and palm line (**1322**)). The system may then determine the angle θ between the palm line, and the touch pad coordinates such as the touchpad x-axis.

FIGS. **16**A-**16**B depict how a simplified exemplary palm angle rotation transformation may help the system relate raw touchpad data to a standard biomechanical and/or anatomical model of the human hand, in accordance with one embodiment of the present invention. The process of rotation transforming the raw touchpad data (x_i, y_i) into palm angle corrected touchpad data (x'_i, y'_i) may be done using the following coordinate rotating formula;

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & -\cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \text{ where } x' = x\cos\theta - y\sin\theta \quad (1)$$

$$\text{and } y' = x\sin\theta - y\cos\theta.$$

The system may find the palm angle using the finger root and/or finger touch points as shown in FIGS. **16**A and **18**.

The system may also calculate the finger touch coordinates (x_i, y_i) (where i=0, 1, 2, 3, 4 for each finger), as well as the palm line rotation angle θ, and the new coordinates (x'_i, y'_i). These calculations may be done using the formula (1) coordinate rotating formula shown above. FIG. **16**A depicts the before the rotation transformation (correction) and the before and after results of this rotation transformation (or correction) and FIG. **16**B depicts the results after the rotation transformation (correction) depicting the palm line direction being substantially parallel to the touchpad's x-axis. It is understood that the word substantially herein refers to an accuracy sufficient to achieve proper guidance of the virtual finger(s) displayed on the display screen to the extent that the user may be able to guide the hand to properly strike a virtual key or other control object displayed on the screen and not intending to imply any more accuracy than so required. Thus, more accuracy is required for smaller virtual keys or control objects than for larger virtual keys or control objects but exact anatomical matching to the hand is not required.

In one embodiment, the system may also determine the finger root (x_{ri}, y_{ri}) location coordinate for one or more of the fingers F**1**, F**2**, F**3**, F**4**. Then the system may perform the analysis often based on the assumption that the F**1** root coordinate (x_{r1}, y_{r1}) is the most available (i.e. most frequently found in the raw touchpad data), which is often true because the finger **1** finger root commonly touches the touchpad surface. Alternatively, because the palm does not bend much, the finger F**1** root coordinate may be calculated from the other palm touch points, i.e. other finger roots (x_{ri}, y_{ri}).

FIG. **17** depicts more exemplary details of the relationship between the finger roots (x_{ri}, y_{ri}). (i.e. roughly finger joint region (**1316**)) and the hand's overall palm angle, in accordance with one embodiment of the present invention. Often there will be missing finger root position data. Here various assumptions, based on the anatomical characteristics of the human hand, may be used to fill in the missing data.

For example, the root coordinates for finger **1** are available (x_{r1}, y_{r1}), then based on hand anatomy considerations, the position of the finger **2** root is likely to be, or may be calculated to be:

$$x_{r2} = x_{r1} + \frac{1}{2}(w_1 + w_2)\cos\theta \text{ and}$$

$$y_{r2} = y_{r1} - \frac{1}{2}(w_1 + w_2)\sin\theta,$$

where w_i is the width of finger i and

$$\frac{1}{2}(w_1 + w_2) = L_{12}$$

as depicted in FIG. **17**.

FIG. **19** depicts how simplified exemplary biomechanical and/or anatomical model data pertaining to the width of the fingers, such as L_{12}, may be used to help interpret raw touchpad data, in accordance with one embodiment of the present invention. A palm line vertical direction (**1930**) may be defined running substantially through touchpad touch point (**1820**) on finger F**2** and substantially perpendicular to palm lines (**1322**). The intersection of palm lines (**1322**) and palm line vertical direction (**1930**) passing through the longitudinal axis of F**2** may pass through the finger root for F**2** at (x_{r2}, y_{r2}), which may be used for the origin of the coordinate rotation axes X, Y. In the same manner, the system may also calculate the likely finger root coordinates for fingers F**3** and F**4** (in this simplified approximation, the model may assume that the finger roots are substantially on the same palm line (**1322**) as per FIG. **16**A, **16**B, **19**, FIG. **13**, and elsewhere). The system may also calculate the new coordinates for any given finger "i" root assuming that the

hand is rotated at palm angle θ by also using rotation formula (1). Here, for example, for finger roots i=1, 2, 3, and 4, the rotation transformed finger root locations may be expressed as:

$$\begin{bmatrix} x_{ri} \\ y_{ri} \end{bmatrix} \xrightarrow{yields} \begin{bmatrix} x'_{ri} \\ y'_{ri} \end{bmatrix}$$

FIG. **20** depicts how in a more accurate exemplary model, the location of the various finger roots will be displaced to some extent from the palm line (which forms the palm angle) by various amounts δri, in accordance with one embodiment of the present invention. Referring simultaneously to FIG. **17** and FIG. **20**, the rotation transformed positions (x'_{ri}, y'_{ri}) of the various finger roots after rotation by palm angle θ may be calculated by the following process. First, calculate the rotation transformed finger **1** root position (x'_{r1}, y'_{r1}) from (x_{r1}, y_{r1}) using formula 1. Second, apply the formulas

$$x'_{r2} = x'_{r1} + \frac{1}{2}(w_1 + w_2) \text{ and } y'_{r2} = y'_{r1} + \delta_{r2}$$

Here often, for a still better approximation, the system may assume that any of finger roots F**2**, F**3** and F**4** might be displaced somewhat from the palm line (**1322**) by a small amount, represented by δ_{ri} as depicted in FIG. **20**. Note that δ_{ri} may be either positive or negative.

Referring simultaneously to FIG. **17** and FIG. **20**, similarly for fingers F**3** and F**4**, the system may use the approximation that the finger roots are usually separated by the width of the fingers to make further approximations or educated guesses such as:

$$x'_{ri} = x'_{r(i-1)} + \frac{1}{2}(w_{(i-1)} + w_i) \text{ and } y'_{ri} = y'_{r(i-1)} + \Delta\delta_{ri}$$

Similarly, for the thumb (finger F**0**):

$$x'_{r0} = x'_{r1} - \frac{1}{2}(w_0 + w_1) \text{ and } y'_{r0} = y'_{r1} + \delta_{r1},$$

where

$$\frac{1}{2}(w_0 + w_1) = L_{01}$$

as depicted in FIGS. **19** and **20**.

Alternatively, using the various rotation transformed finger root coordinates (x'_{ri}, y'_{ri}), the system may also perform the inverse transformation using formula (1) to calculate the raw touchpad data root position coordinates (x_{ri}, y_{ri}) in the original touch pad coordinate system. This later technique is often especially useful for determining if any of the raw touchpad data might represent the thumb root location (x_{r0}, y_{r0}). The raw thumb root touchpad data is often difficult to obtain because sometimes the thumb root does not touch the surface of the touchpad.

Using the techniques described above, the system may make sense of the raw touchpad data by sorting the set of rotation transformed fingertip positions $\{(x'_i, y'_i)\}$ and finger root positions $\{(x'_{ri}, y'_{ri})\}$ according to ascending (or descending) x value, and then attempt to pair the rotation transformed possible fingertip data (x'_i, y'_i) with the rotation transformed possible finger root data (x'_{ri}, y'_{ri}).

Missing Finger Detection Algorithms.

Often the user will have real fingers elevated far enough above the touchpad to produce a "missing finger" problem—that is the touchpad raw data will lack the coordinates of one or more user fingers, and the system software may have to attempt to deduce the existence and location of these one or more "missing fingers".

In one embodiment, a unique touchID may be assigned for each continuous touch. Thus when a finger "i" was previously touched to the touchpad and was lifted later, one may use the touchpad history data obtained by the system at earlier time points (usually a fraction of a second earlier, i.e. time (t-i)) to determine the missing finger. Such time data may also be used in another alternative approach, to be discussed shortly. For example, at time (t-**1**) (i.e. the previous history of stored touchpad data in a time indexed stack of such touchpad data, with fingers F**0**-F**4** identified, one has:

$$\{(x_{0,(t-1)}, y_{0,(t-1)}, \text{touchID}_{0,(t-1)}) \ldots (x_{4(t-1)}, y_{4(t-1)}, \text{touchID}_{4,(t-1)})\}$$

Assume, for example, that currently at time t, the system has a raw set of data for just three touch points from three fingers, such as fingers F**0**, F**1**, F**2** (although this example is using and numbering fingers F**0**, F**1**, F**2**, other fingers and other finger Fi could be alternatively used). The raw data would be:

$$\{(x_{t0}, y_{t0}, \text{touchID}_{t0}) \ldots, (x_{t1}, y_{t1}, \text{touchID}_{t1}) \ldots, (x_{t2}, y_{t2}, \text{touchID}_{t2}) \ldots\}$$

If one finds, for example, that $\text{touchID}_{0,(t-1)}=\text{touchID}_{t0}$ and $\text{touchID}_{3,(t-1)}=\text{touchID}_{t1}$ and $\text{touchID}_{4,(t-1)}=\text{touchID}_{t2}$, then one may tell from the current data set at time "t" that; (x_{t0}, yt_0, touchID_{t0}) belongs to finger **0**, and (x_{t1}, y_{t1}, touchID_{t1}) belongs to finger F**3**, and (x_{t2}, y_{t2}, touchID_{t2}) belongs to finger F**4**. Then one may further determine that fingers F**1** and F**2** are missing, i.e. likely elevated from the touchpad. The positions of various combinations of other fingers may also be analyzed by the same methods.

However at the initial starting time t=**0**, the history data may not be available. Thus, one should determine the missing (e.g. elevated) fingers by one or more various alternate methods, such as the methods described below.

FIG. **21** depicts how the simplified exemplary system may attempt to correlate detected fingertip data from some fingers with finger root data from other fingers, determine that some fingertip data is missing, and thus deduce that these fingers are elevated above the touchpad, in accordance with one embodiment of the present invention. The "missing fingers" include fingers F**1** and F**2**, which are depicted with shading. Missing fingers include a finger that might have been elevated too far above the touchpad in order to register a touch point imprint on the touchpad. To cope with missing fingers the system may operate as follows. First, from the touchpad data set, the system may calculate the palm angle θ coordinates for the various fingertips and finger roots. Second, for each fingertip, the system may check if the position of the fingertip is inside of the range of likely finger root j positions using a formula such as:

$$x'_{rj} - \frac{w_j}{2} \le x'_i \le x'_{rj} + \frac{w_j}{2},$$

where w_i is the width of finger i.

If the fingertip "i" is within this range j, then the system will attempt to match the fingertip "i" with root j. The system, may for example, even attempt to match potential fingertip locations from one finger with the potential finger root data from another finger. For example, the system may attempt to match the fingertip data (x'_1, y'_1) with the finger root position (x'_{r3}, y'_{r3}).

For the thumb finger F**0** root (x'_{r0}, y'_{r0}), and pinky finger (finger F**4**) root (x'_{r4}, y'_{r4}), the range may be calculated as follows:

$$x'_{r0} - length_0 \le x'_0 \le x'_0 + \frac{w_0}{2} \text{ and}$$

$$x'_{r4} - \frac{w_4}{2} \le x'_4 \le length_4,$$

where $length_0$ and $length_4$ correspond respectively to L_0 and L_4 and where $w_0/2$ corresponds to L_B as depicted in FIG. **21**. Note that in this example, the system is also incorporating finger length (i.e. the length between the fingertip (x'_i, y'_i) and the finger root (x'_{ri}, y'_{ri})) into its biomechanical and/or anatomical model of the human hand.

In the frequent cases where finger tips may not be successfully matched with corresponding finger roots, for each un-matched finger root, the system may mark that finger as missing (i.e. likely raised above the surface of the touchpad, rather than touching the touchpad). See for example, fingers F**1** and F**2** in FIG. **21**. Here in FIG. **21**, the shading of fingers **1** and **2** shows that the system has recognized that the fingers tips are missing, probably because the finger tips are elevated a sufficient distance above the touchpad.

Missing Finger Move/Display Algorithms.

In order to show a virtual image of the moving finger, along with fingers that are touching the touchpad, in some embodiments it will also be useful for the system to maintain a stack of the latest n (where n≥1) sets of finger position history information—i.e. retain a history of the most recent finger positions.

In one embodiment, when, as will frequently be the case, the finger position data is insufficient, but the missing finger "i" may be identified by using the previous algorithm or other methods, one may approximate the missing finger's position by assuming that as per a normal biomechanical and/or anatomical hand, the change x and y position of the missing finger's neighboring fingers (e.g. neighboring change Δx and Δy) will also pertain to any change in location of the missing finger as well, as described in the following examples.

Assume that the current time is time "t", and that the locations of the fingers at earlier times (i.e. within a second or a few fractions of a second) have been saved as frames data such as of t-**1** in the stack. The system may compute a weighted mean value such as:

$$\Delta x_{it} = \frac{\sum_{j=1}^{n} (weight_j * (x_{jt} - x_{j(t-1)})}{\sum_{j=1}^{n} weight_j}$$

-continued

and

$$\Delta y_{jt} = \frac{\sum_{j=1}^{n} weight_j * ((y_{jt} - y_{j(t-1)})}{\sum_{j=1}^{n} Weight_j}$$

where j=[1, n] are the touching fingers.

Using the above scheme, then the current position for the missing finger "i" may be calculated as follows.

$$x_{it}=x_{i(t-1)}+\Delta x_{it} \text{ and } y_{it}=y_{i(t-1)}+\Delta y_{it}$$

Note that for (Δx_{it}, Δy_{it}) calculations, one may also use other mathematical methods such as arithmetic mean, median geometrical mean, harmonic mean, and so on.

Finger to Hand Matching Algorithms.

Often, particularly for larger devices, the user may operate the device using two hands. When the user is operating the touch pad with two hands at the same time, the system additionally should be able to decide to which hand the touch points belong to in order to show the user's two hands properly in the display. The system may use various algorithms to help with this decision.

In one embodiment, the system may use the range information on the coordinates after rotating the data by palm angle θ, as is shown on FIG. **21**. In this example, all touch points within the following range may be assumed (i.e. mapped into) one hand. The criteria here may be:

$$\text{For } x[x'_{r0}-\text{length}_0, x'_{r4}+\text{length}_4] \text{ and for } y:[0,\text{length}_2] \text{ or } [0,\max\{\text{length}_0 \ldots \text{length}_4\}]$$

The system may also use the touch angle information for touch points and palm line angles to help assign the raw touchpad data to one hand or the other. Here, for example, the system may assume that both hands belong to the same individual, and essentially extend the biomechanical and/or anatomical model of the human hand to also put in some simplified human anatomical constraints regarding the relationships between the angles of one hand and the angles of the other hand.

FIG. **22** depicts how the simplified exemplary system may further assign raw touchpad data to two different hands (left hand (**2202**) including F**0**L through F**4**L, and right hand (**2204**) including F**0**R through F**4**R) of the same user, based on the assumption that the range of possible hand angles for the same user is limited by the user's anatomy, in accordance with one embodiment of the present invention. According to the previously discussed multi-touch protocol, the touch angle of a touch point may also be determined along the long touch side defined as follows. That is, usually a finger will touch in a roughly oval pattern with the long axis of the oval, i.e. the long touch side, corresponding to the touch angle α of a touch point. For example, based on human anatomical considerations, the angle α between the touch point directions D**4**L, D**2**R and the associated respective palm line vertical direction (**2220**, **2230**) will generally be in the range of [0, 90] degrees. The palm line vertical direction (**2220**, **2230**) is substantially perpendicular to associated palm lines left and right (**1322**, **2222**) respectively. In this example, palm line vertical direction (**2220**) may be associated with finger F**4**L through touch point (**2224**) and palm line vertical direction (**2230**) may be associated with finger F**2**R through touch point (**2234**)

Alternatively or additionally, in one embodiment, the system may also partition the touchpad area (for example,

split the touchpad area into a left half and a right half) and assign some or all of the touch pad data from the left half to the user's left hand, and assign some of all of the touch pad data from the right side of the touchpad to the user's right hand.

Angle Based Finger Matching Algorithms:

FIG. **23** depicts a first simplified exemplary example of angle based finger matching algorithms, in accordance with one embodiment of the present invention. Angle based methods may be used to match raw touchpad data with specific user fingers. These alternative, angle-based, finger matching algorithms may be implemented as follows. First, perform or do a best fit between the touchpad data and the biomechanical and/or anatomical model of the user's hand, and second, use this best fit biomechanical and/or anatomical model of the user's hand, to find a point substantially along a mid-line (**2310**) of middle finger F**2**. Here, for example, one may have mid-line of middle finger F**2** pass through the center of the palm, e.g. palm center point, (x_c, y_c) or other point on the palm (any palm center point (x_{center}, y_{center}) may be used so long as it is inside a region bounded by the five metacarpophalangeal joints (x_{ri}, y_{ri})). The coordinates of the palm center point may be calculated based on the finger model, and known finger positions.

Continuing the above algorithm with the second step, find (e.g. calculate) the finger root (metacarpophalangeal joint) coordinates (x_{r0}, y_{r0}) . . . (x_{r4}, y_{r4}) and calculate the angle α_{ri} of finger root "i" to palm center:

$$\alpha_{ri}=a\tan 2(x_{ri}-x_{center},y_i-y_{center}).$$

The angle calculated by a tan 2 has a range within about $-\pi$ it to $+\pi$. Third, sort the α_{ro}, α_{ri} . . . α_{r4} in ascending or descending order.

FIG. **24** depicts a second simplified exemplary example of angle based finger matching algorithms, in accordance with one embodiment of the present invention. Continuing the above algorithm with the fourth step, find all fingertip coordinates (x_0, y_0), (x_1, y_1) . . . (x_4, y_4) and calculate the fingertip angle α_i to palm center (x_C, y_C)=(x_{center}, y_{center}), where α_i=a tan 2(x_i−x_{center}, y_i−y_{center}). Next, sort α_i in the same order as the α_{ri}. Then, match the corresponding finger to the associated angle, as per FIG. **24**. The advantage of this approach is that one does not need to perform coordinate rotation to match the fingers. Instead, the a tan 2 calculations may be done by computationally faster methods, even by table lookup methods, as needed.

Iterative or Best Fit Embodiments.

In some embodiments, particularly when the raw touchpad data is cluttered or otherwise confusing, the hand and finger analysis software discussed above may operate by an iterative process. For example, the software may make tentative assignments between the raw touchpad data and one possible set of fingertip, finger root, or palm touch points on the previously discussed biomechanical and/or anatomical model of the human hand (here the user's hand), and score the results according to how close the raw touchpad data, either before or after various possible transformations, may fit with a known hand configuration. The software may then explore other possible hand configurations and transformations, and then select or choose the hand configuration and/or transformation (e.g. rotations, translocations, missing fingers, and the like) that produces the highest overall score. The software will then use the highest scoring hand configuration and orientation model for virtual hand display purposes.

Optional Imaging.

In some embodiments, to improve accuracy (that is to replace standard human hand biomechanical and/or anatomical model default parameters with actual user calibration parameters), it will be useful to acquire an image of the user's hands, and to employ various image processing and analysis techniques to analyze this image of the user's one or more hands to better estimate the relative length of the various bones of the user's hands. Indeed, in the event that the user has lost one or more fingers, the system may then use this information to make corresponding changes in its biomechanical and/or anatomical model of the human hand. In other words, the model may include calibration information associated with an image of at least a portion of the hand of the user.

FIG. **14** depicts how the simplified exemplary user's hand or hands may be photographed by the device's camera or other camera, and this image information may be used to refine the default parameters of the biomechanical and/or anatomical model of the user's hand, in accordance with one embodiment of the present invention. In acquiring such images, often it is useful to have the system provide a standardized background, such as a series of distance markings, grid, graph paper, and the like (**1400**) in order to better calibrate the image of the hand and correct for image distortions. This standardized background may additionally include various color, shades of gray, and resolution test targets as well. The background may be conveniently provided by, for example, electronically providing one or more background image sheets (e.g. a jpeg, png, pdf or other image file) for printing on the user's printer.

In one embodiment, the user may put each hand on background (**1400**), and take a photo of the hand(s) (**1402**) with either the computerized device's camera or other camera. This image may then be analyzed, preferably by an image analysis program. The background image will help correct for any image distortions caused by different camera angles, and the like. The user hand image analysis may be done onboard the user's handheld computerized device, but it need not be. In an alternative embodiment, the user may upload one or more images of the hand taken by any imaging device to an external image analyzer, such as a remote internet server. In either event, the image analyzer will analyze the user's skin or hand outline appearance (**1404**), deduce the most probable lengths one or more bones of the user's hand, such as the user's various finger and thumb bones, and send this data or other data to correct the default biomechanical and/or anatomical model of the user's hand(s) back to the user's computerized device, such as for example during calibration step **906** referenced in FIG. **9** above.

Alternatively, at least with more sophisticated and possibly next-generation touchpads capable of providing position details for a large number of contact points, the user may calibration the touchpad by firmly pressing a portion or all of the user's hand on the touchpad, and allowing a highly capable touchpad to in turn precisely render the resulting handprint. A compute program may then analyze the touchpad-derived handprint, extract parameters such as finger joint positions, probable finger and hand bone lengths, and the like and derive the same information as previously discussed for the photographic calibration step above. In other words, the model includes calibration information in accordance with pressing a portion of the hand of the user on the touchpad.

Alternatives or Supplements to the Touchpad.

In an alternative embodiment, information on the user's finger placement may be obtained using optical methods. Thus in an alternative embodiment, the touchpad sensor may be an optical method such as one or more cameras. These camera(s) may keep track of the user's hand and finger positions, and this data may then be fed into the biomechanical and/or anatomical model of the human hand(s) to compute a graphical representation of at least the user's fingers as described previously.

Real Time Video Updating

In another embodiment, image information may also be used to refine the biomechanical and/or anatomical model of the user(s) hands in real time while the user is using the touchpad.

FIG. **15** depicts how an exemplary device camera (**1500**) may be used to obtain a partial image of the user's hand (**1506**) while using the device's touchpad (**1508**), and this information also used to update and refine the biomechanical and/or anatomical model of the user's hand, in accordance with one embodiment of the present invention. The rear mounted device camera (**1500**), which often will have a very limited field of view at close range (**1502**), may nonetheless be used to obtain a real time video image of a portion or part (**1504**) of the user's hand (**1506**) while the user is using a rear mounted touch pad (**1508**) using a touch pad mounted on the back of the computerized device (**1510**). At the same time, touch pad data gives the position of the user's index finger (**1512**) as a strong touch pad signal, and the position of the user's middle finger (**1514**) as a weaker touch pad signal (**1514**).

Note that although the portion of the hand (**1504**) that may be directly visualized by video camera (**1500**) does not include any image information at all pertaining to the position of the user's fingers, the image information (**1504**) does provide a useful series of further constraints upon the biomechanical and/or anatomical model of the user's hands. Thus, the partial hand image information, in conjunction with the touch pad data (**1512**), (**1514**), and optionally with a refined biomechanical and/or anatomical model of this user's hand (if available) obtained in FIG. **14**, above, may improve the accuracy of the depiction of the user's hand and fingers.

In some embodiments, for amusement or artistic purposes, the user may not wish to have a fully accurate anatomical model of the user's virtual hand displayed on the screen, but may instead prefer a variant, such as a realistic depiction of a "monster hand" with fingers being replaced by claws, fur, or pads, and the like, or of a skeleton hand that shows the underlying biomechanical and/or anatomical estimation of the user's hand bones as per FIG. **13**.

In one embodiment, the system software may also be configured to render the user's fingers and hands as various hand variants when displayed. Generally, these hand variants will still provide realistic information pertaining to the user's hand and finger placement, but will also provide this information as various user artistic options that often may be customized according to user preference.

Three Dimensional Multi-Touch Gesture Controls.

Commonly, touchpad controls to a computerized system have focused on two dimensional finger gesture controls requiring finger contact on the locally two-dimensional touchpad surface, even if that surface as a whole may be curved or otherwise project into the third dimension to some extent. In contrast, the embodiments of the present invention, which may operate using a biomechanical and anatomical model of the human hand, may include a three dimensional gesture component that enables various types of three dimensional multi-touch gesture controls described below. Three dimensional multi-touch gesture controls may be advantageous in applications where the user needs to touch a portion of the touchpad continually, such as for example, when the user holds a handheld computerized device including a touchpad on the backside of the device. The three dimensional multi-touch gesture controls may help the computerized system differentiate touches on touchpad control regions intended as control inputs from touchpad touches used to merely hold the device.

In some embodiments, the three dimensional sensing aspects of the present invention may be used to control virtual keyboard data entry to a computerized system by various "lift and tap", or "lift and drag", or "lift and other gesture" type modes for data input. More complex variants can also implement other commands, such as "lift and tap and rotate (with two fingers)", "lift and tap, and enlarge (with two fingers)", and so on.

In one embodiment, the biomechanical and anatomical model of the user's hand may inform the system when one or more user fingers are positioned on the touchpad so as to be above a corresponding control region of the touchpad, such as above a key of a virtual keypad, virtual keyboard, or above a hyperlink, and/or the like, but not yet touching the corresponding control region. Because the model of the hand accurately determines the location of the one or more user fingers even when the user's finger is not touching the surface of the touchpad, the "off-touchpad" finger location may be used for three-dimensional gesture control.

In one embodiment, the control region of the computerized system may be on a touchpad including an integrated display screen located in substantially the same location. For example, integrated touchpads may include both display screen and touchpad built in layers and accessible from the same surface or side of the computerized device and thus located substantially in the same location even though the layers may be separated by small dimensions relative to the touchpad surface length or width. In an alternative embodiment, the control region of the computerized system may be on a separate and/or additional touchpad being located in a location that is different from the location of the display screen as previously described in reference to FIG. **8**. For example, integrated touchpads may include both display screen and touchpad built in layers and accessible from the same surface or side of the computerized device and thus located substantially in the same location even though the layers may be separated by small dimensions relative to the touchpad surface length or width.

In one embodiment, the user moves a finger onto a control region of the touchpad, and this finger is in contact with the touchpad. The computerized system may determine if the user wants to activate that control region, e.g. press the virtual key or control region to generate an input to the computerized system using, for example, a "lift and tap" type control scheme described as follows. When the system does not receive inputs according to the "lift and tap" or "lift and related type" control schemes described below, the initial finger touch, even if touching a control region of the touchpad, may not generate unwanted control inputs enabling the user to continue just safely holding the touchpad.

FIG. **26** depicts a simplified exemplary flowchart of a "lift and tap" technique of key entry for controlling an input from a user to the computerized system, in accordance with one embodiment of the present invention. FIG. **26** depicts the "lift and tap" technique includes obtaining (**2510**) data from

a touchpad, the data being associated with the location and movement of a finger and/or hand of the user and not associated with an image of the finger of the user from an image sensor, when the user operates the computerized system using the touchpad. The "lift and tap" technique further includes communicating (**2620**) the data from the touchpad to the computerized device and analyzing (**2630**) the data in accordance with a model of a human hand, such as referenced in FIG. **9**-FIG. **25**.

FIGS. **27**A-**27**F depict a series of simplified exemplary display screen shots of the "lift and tap" technique of key entry depicted in FIG. **26** being used to type the first two letters of a "Hello World" message on the computerized system, in accordance with embodiments of the present invention. FIGS. **27**A-**27**F were obtained as screen shots or grabs in that respective order from a few seconds of video showing the display screen of a prototype handheld computerized device while the user proceeded to type at touch-typing speed using the "lift and tap" technique referenced in FIG. **26**. The system has already assigned the touch data from the touchpad to at least one of the multitude of fingers of the model, computed a graphical representation of the at least one finger of the user, in accordance with the model, and now displays the graphical representation of the fingers (F**1**, F**2**, F**3**) and hand (**2701**) of the user on a display screen (**2700**) of the computerized system. Note that hand (**2701**) including fingers (F**1**, F**2**, F**3**) is displayed clearly as a virtual, i.e. computer-generated, hand because the palm includes square edges, the fingers include straight sides, and the joints between the fingers and the palm are not continuous.

Referring simultaneously to FIG. **26**, FIG. **27**A, and FIG. **27**D, in one embodiment, the system (optionally) determines (**2640**), in accordance with the touchpad data and the model of the human hand, that at least one user finger (F**1**, F**2**, F**3**) is initially touching, or is in contact with, the region of the touchpad corresponding to a virtual key (**2702**, **2703**, **2705**, **2725**) or other control region.

In one embodiment, when the system first detects that a particular user finger initially touches or is in contact with a virtual key or other control region, the system may optionally generate a graphical representation associated with the control region being touched on display screen (**2700**) of the computerized system. For example, FIGS. **27**A-**27**F further depict the system is generating and displaying on display screen (**2700**) a graphical representation of a virtual keyboard including a multitude of virtual keys, including virtual keys (**2702**, **2703**, **2705**, **2725**), corresponding to control regions on the touchpad.

In one embodiment, the system may then change the appearance of the graphical representation of the touched virtual key or other control region to show or indicate the control region is being initial touched thus providing confirmative feedback to the user. For example, the change of the display image of the touched virtual key may be shown as a change of size, color, shape, slight displacement of the position of the control region image, slight change in display type such as distortion of the control region image, flashing, and/or the like. For example, FIG. **27**A depicts virtual keys (**2702**, **2703**, **2705**), which are being initially touched by respective user fingers (F**1**, F**2**, F**3**) are temporarily displayed including a slightly larger size and a slightly bolder upper border than the remaining untouched keys.

Referring simultaneously to FIG. **26**, FIG. **27**B, and FIG. **27**E, in one embodiment, the user may next lift the at least one user finger (e.g. graphically represented by finger F**2** in FIG. **27**B, and finger F**3** in FIG. **27**E) and the system determines if the now missing in touch contact or lifted finger is most likely positioned above the same previously touched virtual key or other control region on the touchpad. In some embodiments, the system will use the biomechanical and anatomical model of the user's hand to make the above determination. In other words, the system determines (**2650**), using the model that at least one finger of the user (e.g. graphically represented by finger F**2** in FIG. **27**B, and finger F**3** in FIG. **27**E) is positioned above but not touching the control region of the touchpad (e.g. graphically represented by virtual key "H" in FIG. **27**B, and virtual key "E" in FIG. **27**E). The model may be used to determine, for example, that although the touchpad may no longer directly sense that the user's finger is in contact with that particular virtual key or other control region, nonetheless the finger is positioned directly above the control region, in accordance with touch data from other regions of the user's hand and/or fingers and the constraints of the model of the human hand.

It is understood that even if the user initially positions his or her hand and associated fingers in contact with the touchpad such that the at least one finger (e.g. graphically represented by finger F**2** in FIG. **27**B, and finger F**3** in FIG. **27**E) does not initially contact the control region, then the system may still properly determine what control region is positioned directly below the at least one finger not in contact with the control region. In other words, step **2640** described earlier may be an optional step in some embodiments because the model may determine the fingertip locations even with the at least one finger not initially in contact with the touchpad but hovering over the control region using the constraints of model of the human hand.

Still referring simultaneously to FIG. **26**, FIG. **27**B, and FIG. **27**E, in one embodiment, the system may temporarily (for example, a first time interval between 0.05 and 5 or even 10 seconds) change the appearance of the graphical representation of the virtual key or other control region the at least one finger is hovering over. In other words, when the at least one finger is positioned above but not touching the first control region, the system may display the graphical representation of the control region with a different appearance than the prior appearance of the control region. The difference in the graphical representation may be an enlargement of the represented virtual key or other control region, as shown at (**2720**) and (**2730**), or the difference may be another visual change such as a change of size, color, shape, slight displacement of the position of the control region image, slight change in display type such as distortion of the control region image, flashing, and/or the like.

In another embodiment that might be useful for handicapped individuals, the system may instead produce or generate a sound signal audible to the user instead of, or in addition to, the changed appearance of the graphical representation of the control region when the at least one finger of the user is positioned above but not touching the control region. Note that in FIG. **27**B the graphical representation of virtual keys (**2702**, **2705**) remains displayed as in FIG. **27**A because user fingers (F**1**, F**3**) continue to touch the touchpad in FIG. **27**B.

Referring simultaneously to FIG. **26**, FIG. **27**C, and FIG. **27**F, in one embodiment, when the user lowers the at least one finger (e.g. graphically represented by F**2** in FIG. **27**C, and F**3** in FIG. **27**F) back onto or touching the touchpad in that region of the touchpad that corresponds to the control region (e.g. graphically represented by virtual key "H" in FIG. **27**C, and virtual key "E" in FIG. **27**F). In one embodiment, to prevent false inputs and/or to enable finger repositioning without command input, the user's finger low-

ering may optionally be required to happen within a certain time interval or "first period of time" (usually on the order of between 0.01 seconds and 5 or even 10 seconds after the system optionally changes the appearance of that particular key or other control region at the start of step **2650**.

In one embodiment, the system may then verify, using the biomechanical and anatomical model of the user's hand, that the user's at least one finger has now "struck" or "tapped" the particular virtual key or other control region. In other words, the system determines (**2660**) that the at least one finger is subsequently touching the control region in accordance with the data and the model. In one embodiment, the system may record, or register that the appropriate virtual key has been pressed by storing a record of that action in memory. In FIGS. **27**A-**27**C the user is inputting a command to the computerized system to type the letter "H". The system recognizes by the lift and tap action of user finger F**2** that the user is commanding the system to type the letter "H", and the system generates and displays a corresponding letter "H" (**2722**) on display screen (**2700**) to confirm the execution of the command. In FIGS. **27**D-**27**F the user is inputting a command to the computerized system to type the letter "E". The system recognizes by the lift and tap action of user finger F**3** that the user intends to command the system to type the letter "E", and the system generates and displays a corresponding letter "E" (**2732**) to confirm the execution of the command.

In one embodiment, the system may optionally change the displayed appearance of the graphical representation of the struck or tapped virtual key or other control region, often back to either its original appearance (as depicted in FIG. **27**C and FIG. **27**F) or an optional different altered appearance (e.g. a "key struck" appearance) to visually confirm the control region is touched or struck. In one embodiment, the altered appearance may include visual change such as a change of size, color, shape, slight displacement of the position of the control region image, slight change in display type such as distortion of the control region image, flashing, and/or the like. In one embodiment, the optional different altered appearance may be displayed for a short second period of time or time interval, often in the 0.05 to 1 second range, but may extend longer, such as up to 10 seconds. In one embodiment, alternatively or additionally, the system may also generate an auditory signal that the user's actions have resulted in the pressing of a particular virtual key or other control region.

FIG. **28** depicts a simplified exemplary flowchart of a "lift and drag" technique of key entry for controlling an input from a user to the computerized system, in accordance with one embodiment of the present invention. The above embodiments may be extended to other input techniques, such as "lift and drag". In a "lift and drag" technique, in one embodiment, the control region may include an elongated control region with a length substantially greater than the longitudinal length of the surface region of the user's at least one finger when contacting the touchpad. Such an elongated control region may include a virtual slider (e.g. a virtual linear control region), virtual rectangular or circular rotation (e.g. a virtual control knob) or virtual expansion-contraction control region, and the like. In another embodiment, the control region may include a multitude of file names, images, icons, and the like, so that the user may drag these file names, images, icons and the like via the drag technique to, for example, execute a move and/or copy command on the corresponding files or virtual objects.

In one embodiment, after determining that the at least one finger is subsequently touching (**2660**) the first control region in accordance with the data and the model and optionally within a certain third time interval (usually on the order of between 0.0 seconds and 5 or even 10 seconds after the user's finger initially contacts the touchpad), the user may move or slide the at least one finger on the elongated control region of the touchpad. The system may then verify, using the biomechanical and anatomical model of the user's hand, that the user is moving or sliding the at least one finger over the elongated virtual key or other control region. In other words, the system determines (**2810**) that the at least one finger is subsequently touching the first control region in accordance with the data and the model. The system may then store (**2820**) in memory a record of the moving or sliding of the at least one finger (e.g. register that for example, a slider has been moved, and the like), and then optionally change the appearance of the elongated virtual key or other control region, such as file names, images, icons and the like, to visually confirm the command action was executed (e.g. move a knob on a slider control). In one embodiment, alternatively or additionally, the system may also give an auditory signal that the user's actions have resulted in the actuating of the drag command and associated result.

In one embodiment, the user may lift two or more fingers. In other words, the system may determine using the model, that a multitude of fingers of the user are positioned above but not touching the control region of the touchpad. The user may then lower the two or more fingers to the touchpad. In other words, the system may determine that the multitude of fingers are subsequently touching the control region in accordance with the data and the model.

Then in one embodiment, the system may determine a motion of a first finger in relation to a motion of a second finger different than the first finger and assign a command to control the computerized system in accordance with the determined motion. For example the user may either move the two fingers further apart on the touchpad to change the displayed image, e.g. magnify or zoom-in on a displayed image, move the two fingers closer together to zoom-out, or rotate the fingers around a rotation point intermediate between the two fingers to rotate an image. The system may be configured to do corresponding display screen operations under touchpad control where the image on the display screen expands or contracts, or rotates according to the rotation direction of the fingers when the relative motions of the two fingers are assigned to the respective commands. In one embodiment, the system may not require the use of a virtual key or other control regions. Instead, the system may operate as if the entire screen is a virtual control region that may be subject to zoom-in, zoom-out, and/or rotation controlled as described above.

Most existing two dimensional multi-touch gestures may be similarly extended or modified into corresponding three-dimensional counterparts that incorporate the finger lift gesture component described above. Examples of existing multi-touch gestures that may be modified for additional finger lift functionality include various Apple OXS gestures, such as, but not limited to: swipe behind full-screen apps, two-finger scroll, tap to zoom, pinch to zoom, swipe to navigate, open launchpad, show desktop, look up, app expose, rotate, three-finger drag, tap to click, secondary click, notification center, and show web browser tabs.

Distinguishing Between Control Regions and "Holding Regions."

Although users may, for example, use back mounted touchpads to control their various handheld computerized devices, in some situations, users may simply wish to hold

their handheld computerized devices in the same region as the back mounted touchpad, which may create false command inputs when the user inadvertently touches a control region but really intends to merely hold the device by touching the touchpad. In these and related situations, according to one embodiment, the user may designate a portion of the touchpad surface area as being reserved for non-control purposes (e.g. "holding" purposes), hereinafter also referred to as a "non-control" region of the touchpad. In other words, the system enables the user to designate or lock out a portion of the touchpad temporarily as a non-control region for holding the handheld computerized device without controlling an input when the user touches the non-control region.

In one embodiment, the system enables the user to designate some or all of a touchpad as being at least temporarily a non-control region or off limits from a device control perspective by including an actuating button (either real or virtual). Alternatively, in another embodiment, certain user hand gestures, such as a swipe border gesture followed by a swipe "x" gesture within the border, may be assigned and recognized by the system as temporarily turning off touch control within the portion of the touchpad covered by the border and the swiped "x". The user may then safely hold the handheld computerized device or other device by the non-control region of the touchpad. When the user wishes to return to controlling the computerized device using the non-control regions of the touchpad, the user may then, in one embodiment, actuate a corresponding "restore" (real or virtual) button, or implement an appropriate "restore control" gesture or set of gestures designated to execute the restore control command.

Trademarks: iPAD™ and iPhone™ are trademarks of Apple Inc., Cupertino Calif. Surface™ is a copyright of Microsoft Corporation.

In some embodiments, the methods and models discussed above may be enhanced by the use of print information obtained from contact areas on the touch panel. In some embodiments, the print information may include one or more of fingerprints, palm prints, hand prints, and prints generated from other epidermal ridges. For example, the methods and models discussed above may include the use of print information to more accurately achieve the intended results.

In some embodiments, a touch panel is configured to sense epidermal prints or partial epidermal prints caused by patterns of epidermal ridges contacting the touch panel, where the epidermal ridges are spaced apart by epidermal grooves. For example, touch panel **800**, discussed above, may be configured to sense fingerprints, palm prints, hand prints, or other epidermal prints. In some embodiments, the touch panel **800** is configured to sense the prints with a pressure sensitive surface having a sufficient resolution. In some embodiments, the touch panel **800** is configured to sense the prints with a separate fingerprint sensor, such as that sold by AuthenTec, Inc. In some embodiments, the fingerprint sensor may use optical and/or capacitive sensing technology. In some embodiments, the fingerprint sensor may be configured to sense the prints and to also sense points on the touch panel which the user is contacting. In some embodiments, the fingerprint sensor is configured to sense the prints, and a separate sensor, such as a proximity sensor or a pressure sensor is additionally used to sense points on the touch panel which user is contacting. In such embodiments, the pressure sensor may be between the graphics display screen and the fingerprint sensor. Alternative arrangements may also be used.

In some embodiments, a relatively low resolution scan of the touch panel is performed. Based on information from the relatively low resolution scan, contact areas of the touchpad, which are contacted by the user, are determined.

In some embodiments, at least parts of all or substantially all contact areas on the touch panel are additionally scanned, and information representing the prints corresponding to the parts of all or substantially all of the contact areas is transmitted to a processor. The information may represent a relatively high-resolution representation of all of the contact areas. In some embodiments, the additional, high resolution scan is conditionally performed. For example, the high resolution scan may be performed in response to a characteristic of a pattern of contact areas determined with the low resolution scan.

In alternative embodiments, a first portion of the user's skin contacting the touch panel in a first contact area is scanned at a first resolution, and a second portion of the user skin contacting the touch panel in a second contact area is scanned at a second resolution, where the first resolution is higher than the second resolution. Likewise, additional portions of the user's skin contacting touch panel in other contact areas may be scanned at additional resolutions. In some embodiments, certain portions of the user's skin contacting the touch panel is not scanned.

For example, FIG. **29** is an illustration of a hand **2900** contacting a touch panel at contact areas **2910**. In some embodiments, all of the contact areas **2910** are scanned or digitized with a same resolution. In some embodiments, some of the contact areas **2910** are scanned or digitized at a resolution higher than other contact areas **2910**. For example, contact areas **2910***a*, **2910***b*, **2910***c*, and **2910***d* may each be scanned or digitized with a separate resolution. In some embodiments, based on information from an anatomical model and from a proximity or pressure sensor, an anatomical position on the hand **2900** is calculated for each contact area. In addition, a resolution may be selected for each contact area based on the anatomical position calculated therefor. In some embodiments, certain contact areas **2910** are not scanned or digitized, for example, based on or because of the anatomical position calculated therefor.

For example, in some embodiments, contact area **2910***a* may be scanned or digitized with a highest relative resolution, contact area **2910***b* may be scanned or digitized with a next highest relative resolution, and contact area **2910***c* may be scanned or digitized with a lowest relative resolution, where contact area **2910***d* is not scanned or digitized.

In some embodiments, contact areas are repeatedly scanned, for example, periodically. In some embodiments, contact areas are scanned at time intervals dynamically determined. Based on the information of the repeatedly scanned contact areas and based on the anatomical model, movement of a user's hand may be tracked.

FIG. **30** is an illustration of a contact area **2910***a*. As shown, within contact area **2910***a* are ridges **3000** which contact the touch panel within contact area **2910***a*. In some embodiments, all ridges **3000** contacting the touch panel are scanned or digitized with the same resolution. In alternative embodiments, certain portions of the ridges **3000** are scanned or digitized with a first resolution and certain other portions of the ridges **3000** are scanned or digitized with a second resolution, where the first resolution is higher than the second resolution.

For example, a central portion of ridges **3000** may be scanned or digitized with a highest relative resolution, and more peripheral portions of ridges **3000** may be scanned or digitized with progressively lower relative resolutions. In

some embodiments, the most peripheral portions of ridges **3000** may be not scanned or digitized. In some embodiments, the boundary of the central portion may be determined at least partly based on a predetermined shape and size. In addition, a boundary for each next successive additional peripheral portion may also be determined based on predetermined shapes and sizes. Alternative methods for determining portions may also be used. In some embodiments, methods for determining portions, shapes, and sizes may be based at least partly on an anatomical position calculated on the hand. For example, for contact areas corresponding with fingertips, concentric or substantially concentric portions may be used, whereas for contact areas corresponding with the heel of the hand, portions may be bounded by lines which bisect the contact areas.

In some embodiments, scanned print information may be used as an indication of anatomical orientation of the hand. For example, ridges at the end of fingertips tend to be substantially perpendicular to the axis of the corresponding finger. A processor may use this information with the anatomical model to calculate an orientation for the corresponding finger and/or the corresponding hand. Alternatively, the processor may calculate an orientation based on a line generated between, for example, contact areas **2910***a* and **2910***f*, or contact areas **2910***e* and **2910***f*, where contact area **2910***e* may correspond with a gap center of two large palm touch areas **2910***d* and **2910***g*. Once calculated, the orientation information may be stored in a memory. In addition, print information corresponding with other anatomical orientation information may be used by the processor to calculate and store other orientation data.

In some embodiments, scanned print information may be used as an indication of which hand each contact area is associated with. For example, FIG. **31** is an illustration of a pattern of contact areas **3100** generated by fingertips contacting a touch panel. For example, as discussed above, the pattern of contact areas **3100** may be determined using a relatively low resolution scan of the display panel. Absent additional information, the pattern of contact areas **3100**, alone, is insufficient to determine which contact area should be associated with which finger of which hand. For example, as shown in FIG. **32**, the identical contact pattern could be caused by at least the two finger configurations illustrated.

With scanned print information, however, the association of contact areas and fingers is clear. In some embodiments, the scanned print information is collected in response to the pattern of contact areas being insufficient to determine which contact area should be associated with which finger of which hand. In some embodiments, the scanned print information is collected in response to other conditions. In some embodiments, the collection of the scanned print information is not conditioned on contact area configuration. As shown in FIG. **33**, using the scanned print information, the fingerprint ridges indicate that the user's fingers are interleaved. Accordingly, in some embodiments, the scanned print information is provided to a processor with the contact area information. Based on the anatomical model, the scanned print information, and the contact area information, the processor is configured to determine which contact area is to be associated with which finger of which hand. Once determined, the association information may be stored in the memory.

In some embodiments, scanned print information may be used to modify an anatomical model of the hands of the user. For example, certain particular epidermal patterns correspond with particular anatomical features. In some embodiments, such epidermal patterns are recognized or identified by the processor. For example, as discussed above, ridges at the end of fingertips tend to be substantially perpendicular to or to be concave in the direction of the axis of the corresponding finger. Accordingly, an orientation for the corresponding finger may be calculated. In addition, the orientation for the corresponding finger may be calculated based on other information, such as that discussed above. If the orientation calculated based on the print information for the corresponding finger conflicts with orientation calculated based on the other information, modified parameters of the anatomical model may be calculated to resolve the conflict. For example, certain parameters of the anatomical model may be modified so as to accommodate and represent an unusually large bend in the corresponding finger. Once calculated, the modified parameters may be stored in the memory.

In some embodiments, scanned print information may be used to identify a rotation angle of a finger. FIG. **34** is an illustration of a fingerprint **3400** having a highlighted left portion **3410**, a highlighted center portion **3420**, a highlighted right portion **3430**, and a highlighted top center portion **3440**. As shown, the contours of the print information correlate with specific portions of the finger.

Accordingly, based on the scanned print information, an angle of rotation of the finger may be calculated. For example, as shown in FIG. **35**, print information which includes left portion **3410**, indicates that the left portion of finger **3510** is contacting touch panel **3500**. Similarly, as shown in FIG. **29** is an illustration of a hand **2900** contacting a touch panel at contact areas **2910**. FIG. **30** is an illustration of a contact area **2910***a*. FIG. **31** is an illustration of a pattern of contact areas **3100** generated by fingertips contacting a touch panel. FIG. **32**, the identical contact pattern could be caused by at least the two finger configurations illustrated. FIG. **33**, using the scanned print information, the fingerprint ridges indicate that the user's fingers are interleaved. FIG. **34** is an illustration of a fingerprint **3400** having a highlighted left portion **3410**, a highlighted center portion **3420**, a highlighted right portion **3430**, and a highlighted top center portion **3440**. FIG. **35**, print information which includes left portion **3410**, indicates that the left portion of finger **3510** is contacting touch panel **3500**. FIG. **36**, print information which includes center portion **3420**, indicates that the center portion of finger **3510** is contacting touch panel **3500**. Likewise, as shown in FIG. **37**, print information which includes right portion **3430**, indicates that the right portion of finger **3510** is contacting touch panel **3500**. Furthermore, as shown in FIG. **38**, print information which includes top center portion **3440**, indicates that the top center portion of finger **3510** is contacting touch panel **3500**.

In some circumstances, a fingernail line may be additionally or alternatively used to calculate an angle of rotation of the finger. For example, as shown in FIG. **39**, a fingernail line **3450** may be sensed if the user contacts touch panel **3500** with a fingernail.

Because the angle of rotation of the finger corresponds with which portions of the finger, and how much of each portion, contacts touch panel **3500**, based at least in part on an amount of each portion of the finger contacting the touch panel, an angle of rotation of the finger may be calculated. Once calculated, the angle of rotation may be stored in the memory.

In some embodiments, using scanned print information, one or more fingers of the user is characterized, such that print information patterns stored in the memory are associated with specific portions of the one or more fingers.

In some embodiments, scanned print information may be used to identify a rotation angle of a hand. FIG. **40** is an illustration of a hand print **3800** having a highlighted left portion **3810** and a highlighted right portion **3820**. As shown, the contours of the print information correlate with specific portions of the hand.

Accordingly, similar to the angle of rotation calculated for a finger, based on the scanned print information, an angle of rotation of the hand may be calculated. For example, print information which includes left portion **3810**, indicates that the left portion of the hand is contacting the touch panel. Similarly, print information which includes center right **3820**, indicates that the right portion of the hand is contacting the touch panel.

Based at least in part on an amount of each portion of the hand contacting the touch panel, an angle of rotation of the hand may be calculated. Once calculated, the angle of rotation may be stored in the memory.

In some embodiments, using scanned print information, a hand of the user is characterized, so that print information patterns stored in the memory are associated with specific portions of the hand.

The above embodiments of the present invention are illustrative and not limiting. Various alternatives and equivalents are possible. Although, the invention has been described with reference to a handheld computerized device by way of an example, it is understood that the invention is not limited by the type of computerized device or system wherever the device or system may benefit by differentiating between a user's touch on a touchpad for command input and a user's touch on a touchpad for merely holding the device by the touchpad. Although, the invention has been described with reference to certain user fingers touching the touchpad by way of an example, it is understood that the invention is not limited by which user fingers are touching the touchpad. Although, the invention has been described with reference to a touchpad located on the back of a handheld device including a display at the front of the device by way of an example, it is understood that the invention is not limited by where the touchpad is located. Although, the invention has been described with reference to a capacitive touchpad used for data entry by way of an example, it is understood that the invention is not limited by the type of input device. Other additions, subtractions, or modifications are obvious in view of the present disclosure and are intended to fall within the scope of the appended claims.

What is claimed is:

1. A method of processing data during use of a computerized system, the system comprising a touchpad, a processor, and a memory, the method comprising:

obtaining touchpad data from the touchpad, wherein the touchpad data indicates:

a contacted first area of the touch-pad contacted by a first portion of a first hand of a user, and first print information indicating a first pattern of epidermal ridges of the first hand in the contacted first area;

communicating the touchpad data from the touchpad to the processor;

with the processor, analyzing the touchpad data in accordance with an model of a human hand and determining an angle of rotation based on the first print information;

with the processor, associating the first area with a particular anatomical portion of the first hand, wherein the associating is based at least in part on the model, the first area, and the first print information; and

storing the association of the first area with the particular anatomical portion in the memory.

2. The method of claim **1**, wherein the first print information includes ridge location information representing locations of epidermal ridges at first and second spatial resolutions, wherein the first spatial resolution is higher than the second spatial resolution.

3. The method of claim **2**, further comprising selecting a first portion of the first area for the first spatial resolution, wherein the first portion of the first area is selected based at least in part on an model of the first hand.

4. The method of claim **1**, wherein the touchpad data further indicates:

a contacted second area of the touchpad contacted by a second portion of the first hand of the user, and

second print information indicating a second pattern of epidermal ridges of the user in the contacted second area,

wherein the first print information includes ridge location information representing locations of epidermal ridges at a first spatial resolution and the second print information includes ridge location information representing locations of epidermal ridges at a second spatial resolution, wherein the first spatial resolution is higher than the second spatial resolution.

5. The method of claim **4**, further comprising selecting the first spatial resolution for the first area, wherein the first spatial resolution is selected based at least in part on an model of the first hand.

6. The method of claim **1**, wherein the touchpad data further indicates a contacted second area of the touchpad contacted by a second portion of the first hand of the user, and wherein the method further comprises:

determining, based on the model, to obtain the first print information; and

determining, based on the model, to not obtain print information from the contacted second area.

7. The method of claim **1**, further comprising determining an orientation of the particular anatomical portion based on the first print information and on the model.

8. The method of claim **7**, generating a graphical representation of the particular anatomical portion based on the determined orientation.

9. The method of claim **1**, further comprising determining an orientation of the first hand based on the first print information and on the model.

10. The method of claim **9**, generating a graphical representation of the first hand based on the determined orientation.

11. The method of claim **1**, wherein obtaining the touchpad data is performed in response to previously obtained data from the touchpad indicating that the first area is contacted by the user, wherein the touchpad data has a higher resolution than the previously obtained data.

12. A method of modifying a model of a hand during use of a computerized system, the system comprising a touchpad, a processor, and a memory, the method comprising:

obtaining touchpad data from the touchpad, wherein the touchpad data indicates:

a contacted first area of the touch-pad contacted by a first portion of a first hand of a user, and

first print information indicating a first pattern of epidermal ridges of the hand in the contacted first area;

communicating the touchpad data from the touchpad to the processor;

with the processor, analyzing the touchpad data in accordance with a parameterized model of a human hand;

with the processor, modifying one or more parameters of the model based on the first print information; and

storing the modified parameters in the memory.

13. The method of claim 12, wherein the modified parameters are used to calculate at least one of finger joint positions, finger bone lengths, and hand bone lengths.

14. A method of processing data during use of a computerized system, the system comprising a touchpad, a processor, and a memory, the method comprising:

obtaining touchpad data from the touchpad, wherein the touchpad data indicates:

a contacted first area of the touchpad contacted by a first portion of a first hand of a user, and first print information indicating a first pattern of epidermal ridges of the first hand in the contacted first area;

communicating the touchpad data from the touchpad to the processor;

with the processor, analyzing the touchpad data in accordance with an model of a human hand;

with the processor, associating the first area with a particular anatomical portion of the first hand, wherein the associating is based at least in part on the model, the first area, and the first print information;

further comprising determining an orientation of the particular anatomical portion based on the first print information and on the model; and

storing the association of the first area with the particular anatomical portion in the memory.

15. The method of claim 14, generating a graphical representation of the particular anatomical portion based on the determined orientation.

16. The method of claim 14, further comprising determining an orientation of the first hand based on the first print information and on the model.

17. The method of claim 16, generating a graphical representation of the first hand based on the determined orientation.

* * * * *